НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ
імені ІГОРЯ СІКОРСЬКОГО»

факультет інформатики та обчислювальної техніки
(повна назва інституту/факультету)

кафедра автоматика та управління в технічних системах
(повна назва кафедри)

«На правах рукопису»
УДК _____

«До захисту допущено»
Завідувач кафедри
_____ О. І. РОЛІК
(підпис) (ініціали, прізвище)
“___”_____2018 р.

Магістерська дисертація

зі спеціальності (спеціалізації) 126 «Інформаційні системи та технології»
(код і назва спеціальності)

на тему: Ідентифікація користувачів на основі клавіатурного почерку

Виконав : студент __6_ курсу, групи ____ІА–73мп_____
(шифр групи)

Самойленко Олег Станіславович _____ _____
(прізвище, ім’я, по батькові) (підпис)

Науковий керівник доцент, к. т. н. Писаренко А.В. _____
(посада, науковий ступінь, вчене звання, прізвище та ініціали) (підпис)

Консультант _____ _____ _____
(назва розділу) (науковий ступінь, вчене звання, , прізвище, ініціали) (підпис)

Рецензент _____ _____
(посада, науковий ступінь, вчене звання, науковий ступінь, прізвище та ініціали) (підпис)

Засвідчую, що у цій магістерській дисертації немає запозичень з праць інших авторів без відповідних посилань.
Студент _____
(підпис)

Київ – 2018 року

Національний технічний університет України
“Київський політехнічний інститут
імені Ігоря Сікорського”

Факультет інформатики та обчислювальної техніки
(повна назва)
Кафедра автоматики та управління в технічних системах
(повна назва)
Ступінь вищої освіти –другий (магістерський)
(код, назва)
Спеціальність 126 «Інформаційні системи та технології»
(код, назва)

ЗАТВЕРДЖУЮ

Завідувач кафедри

_____ О. І. РОЛІК
(підпис) (ініціали, прізвище)

“___”_____2018_р.

ЗАВДАННЯ

на магістерську дисертацію студенту

Самойленку Олегу Станіславовичу

(прізвище, ім’я, по батькові)

1.**Тема дисертації**__ Ідентифікація користувачів на основі клавіатурного почерку

_____

Науковий керівник дисертації _Писаренко Андрій Володимирович, доцент, к.т.н.

затверджені наказом по університету від “_29_” _жовтня_2018_р.№_____

2.Строк подання студентом дисертації “_4_ ” грудня 2018_р.

3. Об’єкт дослідження: засоби і методи визначення комп’ютерного почерку користувача інформаційної системи

4.Зміст пояснювальної записки**:** а) огляд предметної області; б) розроблення математичних та аналітичних моделей механізму; в) дослідження основних підходів до ідентифікації на основі клавіатурного почерку; г) практична реалізація системи

ідентифікації на основі клавіатурного почерку та дослідження її ефективності; д) розроблення стартап проекту.

5. Перелік графічного (ілюстративного) матеріалу: Діаграма прецедентів; Алгоритм аутентифікації; Алгоритм авторизації; Алгоритм моніторингу; Алгоритм отримання шаблону; Запис даних в кільцевий стек; Алгоритм отримання шаблону; Схема бази даних; Графіки порівняння часу утримання клавіш та часу між натисканням клавіш у п’яти користувачів.

6. Консультанти розділів проекту:

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		Завдання видав	Завдання прийняв

7. Дата видачі завдання “_29_” __жовтня__2018_р.

Календарний план

№ з/п	Назва етапів виконання дипломного проекту	Термін виконання етапів проекту	Примітка
1	Огляд предметної області	02.10.2018	
2	Розроблення моделей розпізнавання	15.10.2018	
3	Дослідження основних підходів до ідентифікації на основі клавіатурного почерку	29.10.2018	
4	Практична реалізація	06.11.2018	
5	Стартап-проект	20.11.2018	
6	Оформлення текстової та графічної документації	2.12.2018	
7	Представлення до захисту	4.12.2018	

Студент _____ Самойленко О. С.
(підпис) (ініціали, прізвище)

Керівник проекту _____ Писаренко А.В.
(підпис) (ініціали, прізвище)

АНОТАЦІЯ

В роботі отримано нове вирішення актуальної практичної задачі підвищення безпеки в інформаційних інфраструктура, а саме ідентифікації користувачів на основі клавіатурного почерку та проведено дослідження основних підходів до ідентифікації користувачів.

Значну увагу в роботі приділено практичній реалізації та дослідженні її ефективності. Результати цього розроблення підтвердили достовірність теоретичних відомостей.

Ключові слова: клавіатурний почерк, динаміка натискання клавіш, ідентифікація, аутентифікація, прихований моніторинг.

SUMMARY

In this work a new solution of the actual practical task of improving security in the information infrastructure, namely, identifying users based on keyboard writing, was obtained, and the main approaches to identifying users were studied.

The considerable attention in the work is devoted to the practical realization and research of its efficiency. The results of this development confirmed the reliability of theoretical information.

Key words: keyboard writing, keystrokes dynamics, identification, authentication, hidden monitoring.

ЗМІСТ

ПЕРЕЛІК СКОРОЧЕНЬ

КП – клавіатурний почерк.

ЧУК – час утримання клавіш.

ІС – інформаційна система.

RSA – абревіатура від прізвищ Rivest, Shamir та Adleman.

СКУД – системи контролю і управління доступом.

FAR – False Acceptance Rate.

FRR – False Rejection Rate.

EER – Energy Efficiency Rating.

ВСТУП

В наш час захист інформації від несанкціонованого доступу виступає вкрай необхідним заходом для запобігання матеріального та нематеріального збитку її власника. Тому дуже важливо брати у розрахунок ефективність роботи підсистеми управління доступом та захисту даних задля збереження безпеки певної системи інформаційної інфраструктури.

Сучасні методи ідентифікації користувача розділяють на три основні групи:

1. Парольні – засновані на унікальній інформації (пароль, пін-код тощо);
2. Атрибутні – засновані на використанні унікального предмету (ключ, токен тощо);
3. Біометричні – засновані на унікальності біологічних та психологічних даних користувача (відбиток пальця, почерк, голос, сітківка ока тощо).

В часи стрімкого розвитку інформаційних технологій парольні та атрибутні способи ідентифікації поступово втрачають свою актуальність, а їх мінуси стають все більш вираженими. Центральною проблемою виступає неточність ідентифікації користувача у системі та велика ймовірність щодо порушення її безпеки в результаті крадіжки, імітації певного атрибуту або злому пароля. Також значним мінусом виступає відсутність функціоналу для виявлення підміни авторизованого легітимного користувача. Тобто, злочинець може незаконно потрапити у систему в момент, коли законний користувач залишає її без контролю після етапу проходження авторизації.

Біометричні характеристики користувача, зокрема клавіатурний почерк (КП), як спосіб аутентифікації, можуть гарантувати підвищений рівень безпеки, неможливості відмови від авторства та комфорт для користувачів, враховуючи невід’ємність біометричних даних від певної людини. Неперервний прихований моніторинг дає можливість своєчасно виявити відсутність законного користувача та перекрити доступ до системи для злодія. Тож на сьогоднішній день стає вкрай

важливим питання вивчення моделей, способів і алгоритмів визначення клавіатурного почерку користувачів інформаційних систем.

Питання вивчення та залучення клавіатурного почерку до алгоритмів ідентифікації та аутентифікації стало центром досліджень багатьох науковців різних країн світу, таких як Д. Сонг, П. Венабл, А. Перріг, Р. Гайнес, В. Лісовський, С. Пресс, Н. Шапіро, А. Пікок, Дж. Леггет, Д. Умфрес, Дж. Вілліамс, Р. Мініханов тощо. В їх працях було запроваджено традиційну для сьогодення схему аутентифікації користувача. Деякі з дослідників біометрії, зокрема А. Іванов, М. Десятерик, В. Марченко, визначили головні властивості КП:

- Швидкість вводу – співвідношення кількості введених символів до часу на їх друкування;
- Динаміка вводу – показники інтервалів між натисканням клавіш та тривалості їх натиску;
- Помилки при введенні тексту та частота їх появи;
- Статистика використання певних клавіш;
- Сила, з якою користувач натискає на клавіші.

Науковці А. Лебедєв, В. Дорохов, Т. Щукін та Є. Луценко в своїх працях визначили та дослідили зв’язок між психофізіологічним станом користувача та періодичних відмінностях у його клавіатурному почерку.

Крім того, дослідники В. Волчихін та А. Іванов запропонували застосувати систему штучної нейронної мережі, здатної здійснювати математичну обробку біометричних даних користувачів.

Вищезазначені методи аутентифікації, на жаль, мають значні проблеми та допускають шанс на появу помилок першого та другого роду в разі визначення клавіатурного почерку з використанням лише ключової фрази.

Слід також взяти до уваги, що клавіатурний почерк – це нестатична біометрична риса людини і може змінюватись залежно від психоемоційного та фізичного стану користувача. Тож програмні реалізації спроб визначення клавіатурного почерку мають досить низький рівень точності ідентифікації і аутентифікації, а також

великий шанс на утворення похибок першого та другого роду. Це призводить до того, що в результаті вони не придатні для прихованого моніторингу клавіатури та виявлення підміни оператора. Отже, в результаті цього виникає актуальність розробки нової системи визначення КП та її програмного забезпечення, що допомогло б зробити ідентифікацію та аутентифікацію користувача більш якісною і точною.

Тож для математичного дослідження часу утримання клавіш (ЧУК), притаманної для клавіатурного почерку користувача, з'являється необхідність у залученні математичної статистики.

Також в ході досліджень виявлено, що базуючись на методі визначення клавіатурного почерку на основі оцінки тривалості утримання клавіш при залученні системи оцінки КП, за довільним текстом з'являється можливість визначити клавіатурний почерк користувача. Це дає можливість реалізувати процес прихованого моніторингу та виявляти підміну законного користувача.

На основі аналізу введення з клавіатури базується метод визначення КП за довільним текстом, що втілюється за допомогою алгоритму визначення клавіатурного почерку за часом утримання клавіш та часом між натисканням клавіш. Тож для постійного прихованого моніторингу клавіатури в ході роботи було створено програмне забезпечення, інтегроване в інтерфейс інформаційної системи.

Мета даної роботи – дослідження особливостей аутентифікації та ідентифікації за біометричними параметрами особистості, зокрема за клавіатурним почерком, а також розробка програмного забезпечення, що здійснює допуск користувача до закритої інформаційної системи.

Основні завдання:

1. Визначення характеристик комп'ютерного почерку.
2. Дослідження алгоритмів, методів, моделей та засобів визначення КП користувача інформаційної системи.
3. Дослідження та створення математичних моделей визначення КП користувача інформаційної системи.

4. Розробка алгоритму дослідження КП користувача за тривалістю утримання клавіш.
5. Розробка алгоритму дослідження КП користувача за часом між натисненням клавіш.
6. Дослідження та створення методів збереження та передачі даних про КП користувача.
7. Розробка способу визначення КП користувача за довільним текстом.

Об'єкт дослідження – засоби і методи визначення комп’ютерного почерку користувача інформаційної системи.

Предмет дослідження – клавіатурний почерк, як метод ідентифікації та аутентифікації користувача.

1 ОГЛЯД ПРЕДМЕТНОЇ ОБЛАСТІ

1.1 Можливості ідентифікації користувача в сучасних інформаційних системах

Одним з основних факторів, які визначають стан захищеності тієї чи іншої системи інформаційної інфраструктури, є ефективність роботи системи управління та надання доступу користувачам і захисту інформації, що там зберігається. Для запобігання істотного матеріального та нематеріального збитку потрібен серйозні міри захист секретної та цінної інформації від несанкціонованого доступу. Головним завданням у проблемі захисту інформації в інформаційних системах від забороненого доступу є завдання розмежування функціональних повноважень. Задача спрямована на запобігання можливості зловмисника зчитувати або модифікувати інформацію, що зберігається. Дії по захисту інформації від несанкціонованого доступу включають:

- недопущення зловмисника до ІС, засноване на засобах розпізнавання користувача;
- створення спеціального забезпечення для захисту інформації;
- використання спеціальних засобів захисту інформації від несанкціонованого доступу.

Виявлено такі основні засоби забезпечення захисту інформації від несанкціонованого доступу:

Законодавчі, організаційні та морально-етичні засоби. Проведений аналіз даних засобів показав, що вони мають низьку надійність без підтримки фізичних, технічних і програмних засобів. Також виявлено, що вони володіють високою залежністю від суб'єктивних чинників, наприклад, від загальної організації роботи на підприємстві або в організації.

Фізичні та інженерно-технічні засоби. Виявленими недоліками є висока вартість, необхідність регулярного контролю і проведення регламентованих робіт, можливість подачі помилкових небезпек.

Апаратні і програмні засоби. Виявлені переваги: надійність, незалежність від суб'єктивних факторів, здатність до модифікації і розвитку, універсальність. Виявлено такі недоліки, які проявляються в різних видах даних засобів: висока вартість, залежність від типу обладнання, недостатня гнучкість.

Одним з напрямків застосування програмно-апаратних засобів є системи контролю та управління доступом. Для успішного функціонування системи контролю та управління доступом до ІС необхідне рішення двох завдань:

1) Зробити неможливим обхід системи управління і розмежування доступу.
2) Гарантувати ідентифікацію користувача, який здійснює вхід до системи.

Зазначені завдання виконуються шляхом проведення наступних процесів контролю і управління доступом, що застосовуються до користувача:

1) Ідентифікацію, тобто, надання суб'єктам доступу індивідуальних та унікальних.
2) Аутентифікацію, тобто, підтвердження автентичності ідентифікації суб'єкта з метою доказу того, що суб'єкт є саме тим, ким він представився.

Таким чином зроблено висновок про те, що забезпечення інформаційної безпеки ключової системи залежить від якості функціонування процесів аутентифікації і ідентифікації користувачів.

1.1.1 Сучасні системи контролю та управління доступом до інформації

Виявлено, що в сучасних інформаційних системах процеси аутентифікації, ідентифікації та доступу до інформації пов'язані, в основному, з процесом авторизації, тобто надання авторизованому користувачу законних прав на доступ до інформації, що обробляється і зберігається в системі. Виділено наступні методи авторизації користувачів, що застосовуються в ІС:

1) Парольні – по контрольної парольній фразі або поєднанню букв і символів.
2) Атрибутні – по унікальному предмету (магнітні картки, смарткарти, USB-токени та інші).

3) Біометричні – за фізіологічними параметрами людського тіла або поведінки людини.

Проведено аналіз сучасної ситуації в області сучасних систем контролю і управління доступом. За даними компанії IDC системи управління ідентифікацією та доступом складають 59% від загального ринку засобів 1Т-безпеки. Дослідження проведене CSI / FBI Computer Crime and Security Survey в 2007 році виявило, що 51% компаній для авторизації користувачів застосовують парольні методи, 35% атрибутні методи і тільки 20% біометричні методи. За даними соціологічного дослідження компанії Unisys 68% клієнтів в світі вважають за краще, щоб банки, платіжні системи, державні органи для ідентифікації використовували біометрію замість паролів і карт. Компанія AtSecurity на початку 2013 року провела опитування серед європейських IT спеціалістів на предмет використовуваних ними технологій авторизації в банківській сфері (рисунок 1.1).

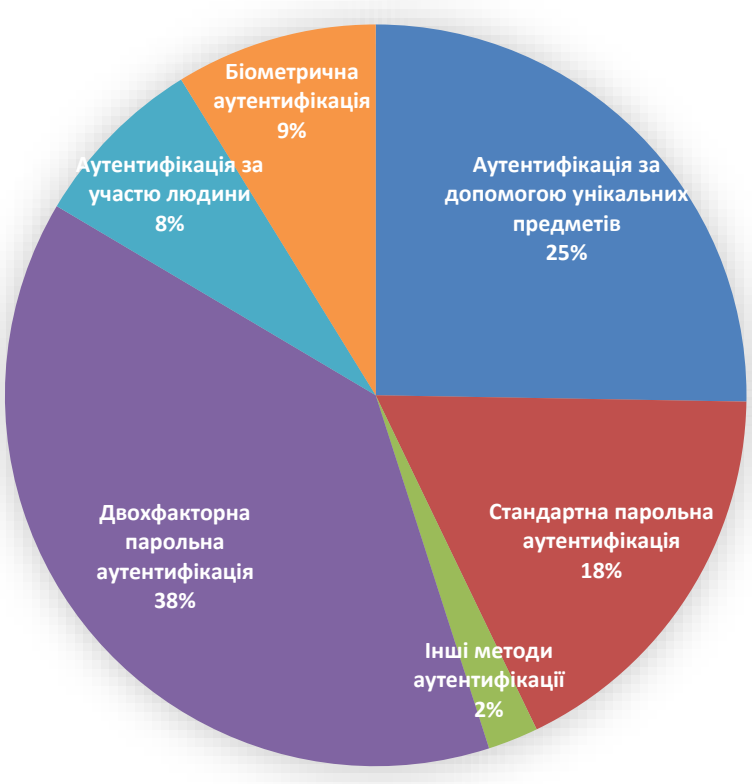


Рисунок 1.1 – Використовувані в європейських банках технологій авторизації за даними опитування AtSecuri

Низька популярність біометричних методів пов'язана з високою вартістю і складністю налаштування біометричних систем захисту інформації.

За підсумками аналізу наведених вище фактів, зроблено висновок про те, що найбільш поширеними є парольні та атрибутні системи контролю та управління доступом. Але вони мають ряд суттєвих недоліків.

1.1.2 Переваги та недоліки парольних систем контролю доступом до інформації

Парольні системи контролю та управління доступом є найбільш часто-використовуваними засобами захисту інформації. Популярність пояснюється тим, що використання даної системи є значно простішою в порівнянні з іншими, але подібні системи мають невисокий рівень безпеки, в зв'язку з наявністю великої кількісті недоліків:

1) Можливість підбору пароля.
2) Невиконання інструкцій по створенню безпечного пароля користувачем (недбале ставлення до процедури вибору пароля).
3) Існування і наявність у вільному доступі спеціалізованих додатків для підбору і злому паролів.
4) Пароль може бути отриманий шляхом застосування насильства до користувача.
5) Пароль може бути вкрадений, тобто перехоплений при введенні власником.

Проведено дослідження надійності застосування парольних методів ідентифікації і аутентифікації. Нижче перераховані виявлені факти, що підтверджують недоліки сучасних методів контролю та управління доступом.

Згідно з даними, вказаними в звіті дослідницької лабораторії RSA:

– 25% користувачів зберігають паролі у вигляді звичайного тексту на комп'ютері;
– 22% використовують для зберігання паролів мобільний телефон;
– 18% записують паролі до облікових записів на листочках паперу.

Експерти з компанії Індід (indeed-id.ru) наводять приклад зберігання паролів до облікових записів користувачів в одній з банківських установ, де компанія проводила оцінку стану інформаційної безпеки. Для швидкого доступу до власних паролів від декількох інформаційних систем використовуваних в банку, більшість співробітників використовували липкий листочок паперу як засіб зберігання даних та приклеювали його до клавіатури.

За підсумками дослідження методів парольної аутентифікації зроблено висновок про неможливість застосування парольних методів для виявлення підміни законного авторизованого користувача. Періодичні введення пароля для підтвердження авторизації під час роботи відволікає користувача від виконання призначених завдань. Відсутня можливість виявити мінімально і максимально допустимі часові інтервали між введенням пароля.

Дослідження атрибутного методу аутентифікації за допомогою унікального предмета дозволяє забезпечити більш надійний захист інформації, ніж парольний. Але атрибутна аутентифікація, як з «пасивними», так і з «активними» унікальними предметами володіє декількома недоліками:

1) Можливість крадіжки предмета у користувача.
2) Необхідність в спеціальному обладнанні для роботи з магнітними картками, смарт-картами та іншими.
3) Можливість виготовлення копії унікального предмета.
4) Можливість підробки унікального предмета.

Проведено аналіз програмного забезпечення, яке блокує доступ до ключової системі в разі вилучення ключа (спеціально запрограмованого USB flash-накопичувача). Розробниками пропонується користувачу забирати USB-ключ з собою, коли від відлучається від робочого місця. В цьому випадку відбувається автоблокування комп'ютера до моменту подальшого підключення ключа. Неможливість підробки та копіювання ключа забезпечується застосуванням алгоритму шифрування AES-256 і захистом ключа PIN-кодом. Людський фактор являється основним недоліком даних систем для захисту від підміни законного

користувача, наприклад користувач може відлучитися від робочого місця забувши вийняти USB-ключ.

1.1.3 Сучасні біометричні системи контролю доступу і можливість їх застосування для виявлення підміни авторизованого користувача

Проведено дослідження біометричних систем контролю і управління доступом. Біометричні СКУД засновані на розпізнаванні фізіологічних і поведінкових характеристик людини. Дані системи класифікуються залежно від виду характеристики людини, що розпізнається (рисунок 1.2).

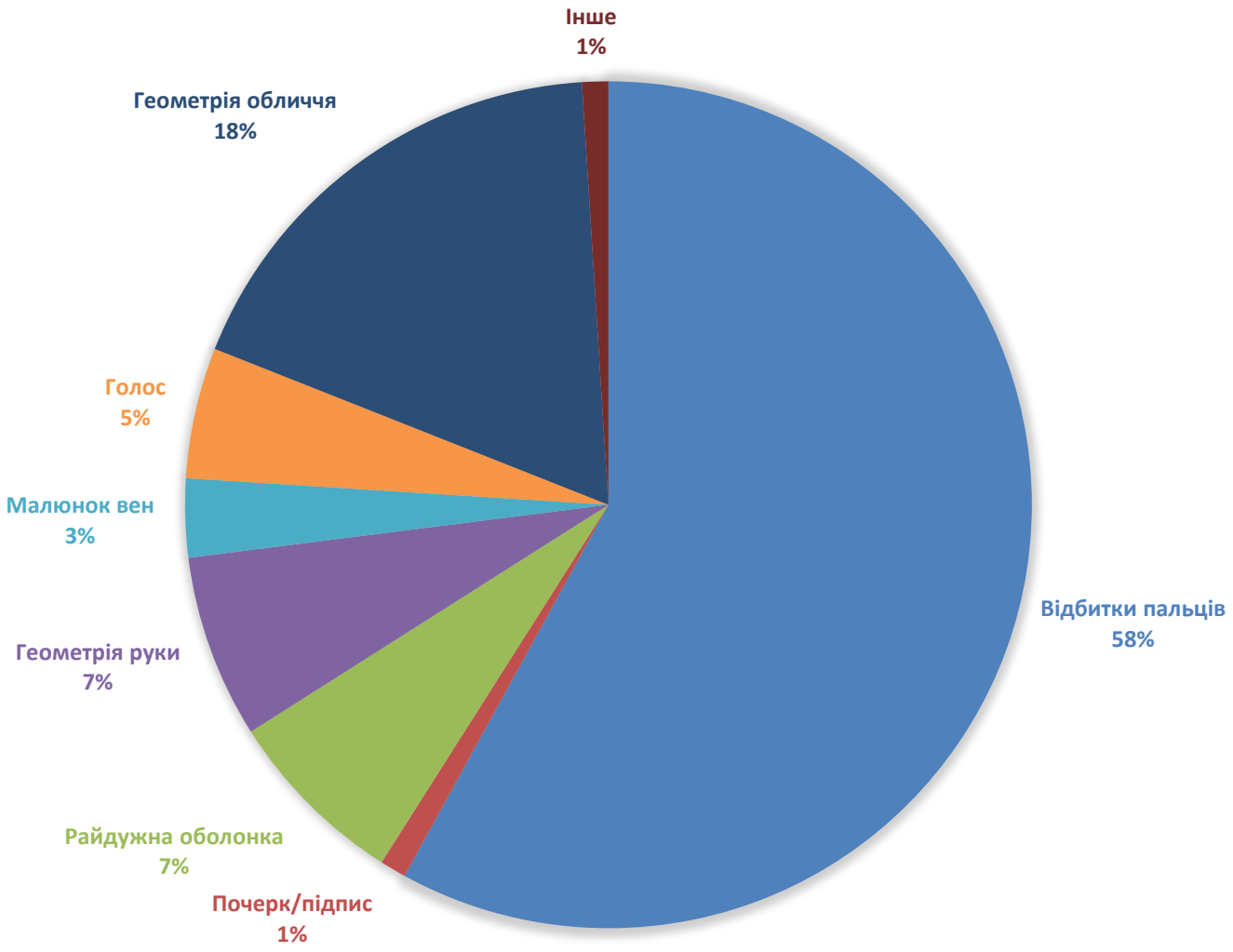


Рисунок 1.2 – Біометричні технології, експлуатовані в промислових масштабах

У процесі аутентифікації пред'явлений користувачем зразок порівнюється зі створеним раніше шаблоном, де ураховується деяка похибка. Вона залежна від необхідного оптимального співвідношення помилок невірних прийнять (FAR) і помилкової відмови (FRR), які відповідають точності і надійності роботи системи.

Проведено дослідження та аналіз існуючих і активно експлуатованих біометричних систем ідентифікації і аутентифікації користувачів і проведена оцінка можливості їх застосування для виявлення підміни авторизованого законного користувача. Результати дослідження представлені в таблиці 1.1.

За підсумками проведеного аналізу зроблено висновок про те, що біометричні системи розпізнавання відбитку пальця, райдужної оболонки ока, геометрії руки, вен руки, геометрії особи людини малозастосовні для виявлення підміни законного оператора, так як мають істотні обмеження у використанні і вимагають виконанню певних умов при скануванні характеристик. Біометричні системи розпізнавання сітківки ока не дозволяють проводити постійний моніторинг особистості користувача, так як вимагають виконання певних умов для сканування сітківки ока.

Таким чином, зроблено висновок про те, що для виявлення підміни законного користувача необхідно використовувати ті біометричні параметри, які проявляються при виконанні користувачем завдань, пов'язаних безпосередньо з його роботою. Найбільш часто користувачем виконується робота з мишею і набір текстів на клавіатурі. Виходячи з даного припущення зроблено висновок про те, що найбільш зручним для забезпечення процедури постійного таємного моніторингу з метою виявлення підміни користувача є клавіатурний почерк – динамічна поведінкова біометрична характеристика людини.

Таблиця 1.1 – Аналіз основних біометричних СКУД

Біометричний параметр	Ціна пристрою (дол.)	Імовірність помилки FAR,%	Переваги	Недоліки	Можливість застосування для виявлення підміни авторизованого оператора
Відбиток пальця	100	0.001	1. Висока достовірність. 2. Стійкість параметра. 3. Малий ідентифікаційний код. 4. Компактний зчитувач. 5. Низька вартість. 6. Застосування додаткових датчиків (температури, сили натискання).	1. Безпосередній контакт з обладнанням. 2. Складність алгоритмів. 3. Легкість пошкодження папілярного візерунка пальців, що ускладнює ідентифікацію. 4. Значна залежність якості зчитування від стану шкіри. 5. Можливість підробки відбитка пальця.	Проводиться впровадження сканерів відбитків пальців у мишки і клавіатури, в корпуси ноутбуків, але більшість з них служать тільки для забезпечення процесу авторизації (наприклад, BioLink U-Match Mouse компанії BioLink Technologies). Застосування з метою виявлення підміни оператора ускладняються необхідністю постійного безпосереднього контакту пальців зі зчитувачем, що є неможливим.
Райдужна оболонка	> 500	0.00001	1. Стійкість параметра.	1.Складність алгоритмів. 2. Висока вартість.	Ускладнюється необхідністю постійного напрямку погляду

ока			2. Висока точність. 3. Надзвичайна складність підробки. 4. Відсутність безпосереднього контакту з обладнанням. 5. Висока швидкодія. 6. Сканування можна проводити на відстані від декількох сантиметрів до декількох метрів.	3.Низька доступність високих рішень.	оператора в бік камери, яка володіє малими кутами сканування. Пристрій EyeLock компанії Hoyos Group розроблено для забезпечення процесу авторизації.
Геометрія руки	> 600	0.2	1.Стійкість параметра. 2.Простота алгоритмів.	1.Безпосередній контакт з обладнанням. 2.Незручна процедура сканування. 3.Великі розміри зчитувача.	Постійний моніторинг неможливий, якщо руки оператора розташовані поза зоною дії сканера.
Сітківка	4000	0.000001	1. Незмінність	1.Складність зчитування.	Відсутня в зв'язку з необхідністю

ока			параметра з плином часу. 2. Висока точність. 3.Відсутність безпосереднього контакту з обладнанням.	2.Складність алгоритмів. 3.Високий час обробки шаблону. 4.Висока вартість системи.	виконання певних умов для зчитування характеристики.
Геометрія особи	>100	От 5 до 0.0047	1. Можливість безперервної аутентифікації. 2. Відсутність безпосереднього контакту з обладнанням. 3. Низька вартість.	1. Залежність від умов освітлення, положення голови. 2.Залежність від міміки обличчя. 3.Залежність від перешкод (окуляри, головний убір, зміна зачіски).	Можливість застосування для постійного моніторингу підміни оператора є, але існує ряд обмежень, спричинених недоліками методу. Основне застосування – процес авторизації, наприклад, контролер-зчитувач STFR.040EM марки Smartec.
Вени руки	>300	0.0008	1. Висока точність. 2.Відсутність безпосереднього контакту з	1.Чутливість сканера до природного і штучного освітлення. 2.Характеристика	Постійний моніторинг неможливий, якщо руки оператора розташовані поза зоною дії сканера. Біометричний зчитувач вен долоні

			обладнанням. 3.Прихованість характеристики.	залежить від стану здоров'я кровоносної системи людини.	PalmVein, представлений компанією Fujitsu застосовується для розпізнавання оператора в процесі авторизації.

1.2 Проблеми виявлення підміни законного користувача

Проведене дослідження загроз виявило, що найчастішими і небезпечними, з точки зору розмірів збитків, є внутрішні загрози, що надходять безпосередньо від співробітників, які мають доступ до системи, а не від зовнішніх для організації зловмисників. До них відносяться: основний персонал, представники служби безпеки, допоміжний персонал, технічний персонал. Основними причинами незаконних дій персоналу є: образа, помста, бажання отримати матеріальну вигоду та інші. Загроза несанкціонованого доступу до інформації внутрішніми суб'єктами ускладнюється тим, що вони знайомі зі структурою та основними функціями, принципами роботи засобів захисту інформації, що застосовуються та мають можливість доступу до систем зберігання інформації.

У дослідженні приділено увагу небезпеки та ймовірності виникнення внутрішніх та інсайдерських загроз інформаційній безпеці. Datapro Information Services Group провела опитування серед випадково обраних менеджерів інформаційних систем з метою з'ясування ситуації в області захисту інформації. Було отримано 1153 анкети, на основі яких отримані наведені нижче результати:

- 3% – зовнішні порушення (наприклад, атаки хакерів);
- 70-75% – внутрішні порушення, з них:
  a) 10% здійснені скривдженими і незадоволеними співробітниками;
  b) 10% здійснені з корисливих мотивів персоналом системи;
  c) 50-55% результат ненавмисних помилок персоналу та користувачів системи в результаті недбалості, халатності або некомпетентності.

Дослідження, проведене компанією InfoWatch в 2006 році, присвячене проблемам інформаційної безпеки російського держсектора охопило 191 державну організацію. Даний проект уточнює результати третього щорічного дослідження «Внутрішні IT загрози в Росії 2006», в ході якого було опитано 1450 російських організацій у всіх сферах економіки. Результати дослідження показали, що зовнішні загрози (45%) зустрічаються рідше, ніж внутрішні загрози (55%). Тільки 15%

організацій вдалось уникнути витоків інформації протягом року. Також компанія повідомляє про зростання загального числа витоків інформації з організацій на 16% в порівнянні з 2011 роком. У звіті «Trends in IT Security Threats: 2007», підготовленому Computer Economics, перше місце по фінансовим збиткам від порушень інформаційної безпеки займають загрози, що надходять від інсайдерів, випереджаючи інші види небезпек. За даними фахівців CSI в 2007 році з погрозами від інсайдерів зіткнулися 59% організацій.

Дослідження, що було проведене кадровим холдингом АНКОР, виявило, що 22% росіян користуються службовою інформацією для стороннього заробітку. Аналіз інформаційної безпеки, проведений компанією Searchlnform в російських організаціях визначив такі факти:

- 19,2% співробітників готові відмовитися від пропозиції про продаж секретної інформації;
- 43,7% співробітників готові заробити на продажу важливої інформації;
- 12% співробітників готові надати конфіденційну інформацію зловмисникам безкоштовно.

За даними опитування, проведеного в 2011 році фірмою Sailpoint Technologies на території США, Великобританії та Австралії показало, що з 3,5 тисячі опитаних співробітників, кожен з яких, по суті, являється інсайдером, досить велика кількість готові вкрасти секретну інформацію у своїх компаній: 22% в США, 29% в Австралії і близько 50% опитаних у Великобританії. 5% респондентів в США, 4% в Австралії і 24% у Великобританії продали б секретну інформацію своєї компанії з метою особистого збагачення. Схоже дослідження, проведене корпорацією Symantec спільно з Профессіонали.ру на території Російської Федерації показало, як співробітники компаній поводяться з внутрішньою інформацією. За результатами проведеного опитування, близько 70% працівників крадуть ділову інформацію, а 56% готові вкрасти інформацію з атрибутами обмеженого доступу. У процесі дослідження аналітики виявили чотири типи співробітників-інсайдерів:

- 24% можуть піддати компрометації корпоративну мережу, не підозрюючи про

це;

- 22% ігнорують базові вимоги безпеки, при цьому усвідомлюють ступінь загрози;
- 7% увійшли до групи тих, хто переслідує власні корисливі цілі;
- 47% опитаних службовців досить обережно поводяться з комерційною таємницею.

Виявлено наступні факти про значний збиток, завданий інсайдерами:

- жителі Америки втратили понад 929 мільйонів доларів через попадання паролів платіжних карт в руки третіх осіб;
- Жером Кервьєль завдав інвестиційному банку Societe Generale, в якому він працював трейдером, збиток в 5 мільярдів євро, використовуючи в своїх махінаціях паролі колег;
- Джагмен Чан, співробітник банку HSBC, вкравши паролі колег, перевів 90 мільйонів євро на рахунки інших банків;
- американська паливно-газова корпорація Enron Corporation оголосила про своє банкрутство через витік інформації, організований її співробітниками.

За підсумками дослідження зроблено висновок про те, що внутрішні загрози є найбільш небезпечними для інформаційної безпеки та часто реалізованими при використанні парольних і атрибутних методів аутентифікації і авторизації в якості засобів захисту інформації. Рішення задачі виявлення підміни користувача є актуальним на сьогоднішній день. Для захисту ІС від несанкціонованого доступу винаходяться різні методи. Наприклад, USB-клавіатура KSI SonarLocID Keyboard, створена для забезпечення збереження важливої інформації. У клавіатуру вбудований радар, який автоматично визначає, чи знаходиться користувач за комп'ютером. Якщо користувач не заблокував доступ до системи, відлучившись від комп'ютера, дана клавіатура за допомогою радара це визначить і сама заблокує систему. Мінусом цієї системи є те, що є можливість її обдурити як випадково, так і навмисно, наприклад, поставивши перед радаром будь-який предмет. Таким чином, проблема виявлення підміни авторизованого користувача залишається актуальною і

вимагає розробки нових варіантів розв'язання проблеми.

1.3 Застосування біометричної ідентифікації в освітніх установах

У сфері освіти біометрична ідентифікація не дуже широко розповсюджена. У переважній більшості випадків використовуються сканери відбитків пальців. Вони застосовуються у шкільних автобусах Каліфорнії при посадці і висадці пасажирів. В

сукупності з GPS- датчиком це дозволяє простежити місцезнаходження дитини. У разі, якщо юний пасажир бажає вийти не на своїй зупинці, водій отримує повідомлення про позаштатну ситуацію і, при необхідності, розбирається в ній.

У шкільних їдальнях, коли оплата обраної страви відбувається за допомогою списання грошових коштів з особового рахунку учня. Впровадження технології дозволяє істотно знизити час обслуговування, а також виключає можливість шахрайства, коли хтось купує по чужій смарт-карті або чужому цифровому паролю.

У деяких школах Британії дана система модернізована так, що контролює раціон харчування дитини. В комп'ютер в їдальні завантажується інформація про продукти, які дитині купувати не можна (через його алергії або надмірної ваги). Столовий комп'ютер щодня запам'ятовує, які продукти вибирав школяр на сніданок, батьки можуть запросити їх список.

На прохідній сканування відбитків пальців застосовується багатогранно. Для безпеки студентів в Караганді, не допускаються до будівлі незареєстровані люди. Якщо даних про людину немає в базі, то йому видається тимчасовий пропуск, відзначається час входу і виходу, а також факт входу і виходу фіксується відеоапаратурою.

Для контролю відвідуваності учнів, зокрема, в ряді медичних ВНЗ адміністрація додатково враховує час, проведений в лабораторіях, для впевненості в достатній освоєнні необхідного матеріалу.

Для обліку робочого часу система отримала особливого поширення в індійському штаті Біхар, де викладачі вписували один одного в журнал відвідувань, а самі не з’являлись на заняття.

Для розмежування доступу до спортивних центрів зонам відпочинку розваг замість карткової системи в університетах Амстердама та Флориди. Застосовується щоб не допускати в зазначені місця тих, хто не має на це права.

В університеті Трої, штат Алабама, набув поширення пристрій, який об'єднує в собі сканер відбитків пальців, мікрофон та відеокамеру. Застосовується він в процесі онлайн іспитів. Перед початком іспиту студент повинен засвідчити свою особистість дотиком сканера відбитків пальців; під час самого іспиту мікрофон і камера відстежують ознаки «підозрілої активності», наприклад, коли у студента раптом змінюється голос або фіксуються сторонні руху. Недоліком такого методу

є вартість обладнання і те, що не кожен захоче бути під пильною увагою відеокамери.

Ще один варіант застосування аутентифікації в освіті використовується центром Pearson Language Tests. Під час тестування, учні проходять ідентифікацію по малюнку вен на долоні, а в день тестування також здійснюється цифрове фотографування кожного з учасників. Всі ці заходи робляться для того, щоб виключити саму ймовірність того, що тест здасть одна людина, а результати, що підтверджують необхідний рівень знання, отримає хтось інший.

Аналіз існуючого стану ситуації показав, що застосування біометричних технологій в сфері освіти в цілому не поширене.

1.4 Огляд методів прихованого моніторингу

Серед біометричних методів прихованого моніторингу можна виділити наступні:

Ідентифікація по геометрії лиця особи заснована на індивідуальних особливостях. Ця технологія максимально наближена до того, як люди взнають один

одного, подивившись на обличчя. Після отримання фотографії через камеру алгоритм будує модель особи, і порівнює її з наявними в базі.

Перевагами використання такого методу буде те, що:

- наявні в особовій справі осіб фотографії можуть бути зразком для створення еталона порівняння;
- зображення можуть бути отримані без будь-якого контакту з людиною, тобто на відстані;
- технологія вважається найменш нав'язливою;
- контроль за допомогою розпізнавання осіб можна здійснювати в процесі будь-якої діяльності користувача в системі дистанційної роботи: при вивченні матеріалів, виконанні завдань, спілкуванні з іншими користувачами.

Проблеми:

- можуть бути проблеми з законодавством через втручання в приватне життя;
- через налаштування безпеки в операційній системі людини, може знадобитися дозвіл користувача на використання камери, що ставить під загрозу скритність даного методу;
- У користувача може не бути камери або ж він може користуватися різними комп'ютерами для роботи чи навчання, в тому числі без камери. Отже, не завжди його можливо буде встановити.

Обмеження:

Більшість алгоритмів розпізнавання чутливі до положення особи та дають збій при нахилах, поворотах, а також при різній емоціональності на обличчі. Різна освітленість так само впливає на якість розпізнавання. Мінімум раз на два роки необхідно оновлювати еталонну інформацію через вікові зміни. Якщо студент носить окуляри, то його розпізнавання за допомогою по геометрії особи стає практично неможливим, так як через відблиски складно визначити опорні точки, тобто центри очей.

Ідентифікація по голосу заснована на поведінкових і біологічних компонентах. Відмінності в тому, як звучать голоси різних людей, залежать від

поєднання біологічних відмінностей в будові голосового апарату і від індивідуальних особливостей мови. Технологія ідентифікації на основі цих відмінностей створює голосову модель, яка буде порівнюватися з даними що були надані. При цьому шаблон може будуватися на проголошенні парольної фрази або ж на тембрі голосу в цілому.

Переваги методу в його надійності, гнучкості та хороших показниках точності. Системи ідентифікації по голосу, як правило, прості в використанні і не вимагають спеціального обладнання та його налаштування, отже, недорогі у використанні.

Проблеми:

- можливість відсутності мікрофона;
- через налаштування безпеки в операційній системі користувача, може знадобитися його дозвіл на використання мікрофона, що ставить під загрозу скритність даного методу.

До обмежень відноситься наступне: голос людини може змінюватися під впливом часу, хвороби або емоційного стану. До того ж при реєстрації еталона на мікрофоні високої якості можуть виникнути труднощі з ідентифікацією через мікрофон більш низької якості.

1.5 Огляд існуючих систем прихованого клавіатурного моніторингу

Витоки аналізу клавіатурного почерку беруть початок в ті часи, коли широке поширення мали радіозв'язок з використанням Абетки Морзе. Досвідченого радиста можна розпізнати по швидкості, своєму стилю і якості передачі сигналів, на відміну від початківця. Так само, як і любитель музики може дізнатися виконавця пісні на слух. Приблизно ті ж самі принципи покладені в основу розпізнавання людини, яка працює на клавіатурі комп'ютера.

Безкоштовний додаток «Клавіатурний почерк 1.0». Він призначений для підрахунку різних характеристик клавіатурного почерку: швидкість введення

символів, швидкість натискань, відсоток помилок. Недоліком є те, що вона не зберігає дані по клавіатурних почерках декількох користувачів, тому немає можливості провести порівняння для ідентифікації користувача.

Аутентифікації за допомогою клавіатурного почерку притаманний ряд переваг і недоліків. До переваг відносяться:

- стабільність клавіатурного почерку конкретного користувача, що дозволяє з більшою вірогідністю ідентифікувати користувача, працюючого з клавіатурою;
- відносна дешевизна впровадження системи розпізнавання;
- можливість контролювати як доступ до ресурсів, так і фізичний стан співробітника.

До недоліків методу можна віднести:

- використання можливо тільки для розпізнавання користувачів зі сформованим клавіатурним почерком.

Можливості аналізу клавіатурного почерку дозволяють вирішувати такі завдання:

- контролювати фізичний стан користувачів;
- відмовитись від використання паролів;
- надати користувачам більш простий спосіб входу в мережу.

Крім того, на сьогоднішній день це єдина технологія, яка може використовуватися за двома призначеннями:

- для ідентифікації користувача, який претендує на доступ комп'ютерній системі;
- для проведення таємного клавіатурного моніторингу працюючих користувачів.

Системи, що вирішують ці завдання, різняться тим, що в першому випадку ідентифікація користувача здійснюється по короткій парольній фразі, а в другому випадку – за довільним текстом.

2 РОЗРОБЛЕННЯ МАТЕМАТИЧНИХ І АНАЛІТИЧНИХ МОДЕЛЕЙ МЕХАНІЗМУ РОЗПІЗНАВАННЯ КЛВІАТУРНОГО ПОЧЕРКУ

2.1 Аналіз клавіатурного почерку в процесах аутентифікації, ідентифікації та виявлення підміни користувача

Проведено дослідження методів ідентифікації і аутентифікації по клавіатурному почерку. Використання методів поведінкової біометрії, заснованої на клавіатурному почерку, на відміну від методів фізіологічної біометрії не вимагає придбання додаткових пристроїв. Для отримання зразка клавіатурного почерку достатньо наявність звичайної стандартної клавіатури. Це робить даний метод недорогим і ненав'язливим для користувача та може бути застосовано таємно, що дозволить покращити існуючі комп'ютерні системи з надання безпеки інформації.

Відзначено, що в основу досліджень клавіатурного почерку покладена робота користувачів телеграфу. В середині 19-го століття, коли телеграф часто використовувався, було помічено, що оператори телеграфу могли ідентифікувати інших операторів по їх ритму набору. Метод «Fist of the Sender» використовувався під час другої світової війни для ідентифікації відправника повідомлення за ритмом, темпом і часом натискання телеграфного ключа. Bryan і Harter провели ряд експериментів на тридцяти семи телеграфних операторах, що мають різні ступені уміння набору. Вони відзначили, що телеграфні оператори змогли дізнатися інших операторів по стилю набору.

Проведено аналіз досліджень в області розпізнавання клавіатурного почерку. На початку 80-их років Національний науковий фонд і Національне Бюро Стандартів в Сполучених Штатах провели дослідження, які встановили, що зразки клавіатурного почерку містять унікальні особливості, які можуть бути ідентифіковані. Shaffer показав, що набір тексту є запрограмованим умінням і рухи під час введення організовуються до їх фактичного виконання. Іванов А.І. в своїх працях показав, що при наборі тексту на клавіатурі однією рукою виявляються задіяні близько 50 м'язів пальців руки і пальців передпліччя і ще приблизно 20 м'язів

плеча і плечового поясу, тобто при друку двома руками людина управляє приблизно 140 м'язами. Тому зразок почерку людини є поведінкової особливістю, яка розвивається з плином часу і, таким чином, не може бути змінена, втрачена або забута. У будь-якій поведінковій біометричній характеристиці можуть спостерігатися великі зміни в особливостях характеристики. Однак, вони повинні надати достатньо інформації, щоб ідентифікувати і визначати справжність особистості за шаблоном почерку.

Купер був першим дослідником машинописного набору тексту, який розбив процес набору тексту на чотири стадії:

1) Сприйняття людиною тексту.
2) Збереження його в пам'яті.
3) Перенесення людиною збережених в його пам'яті символів в команди, що передаються м'язам.
4) Безпосередньо набір тексту з процедурою зворотного зв'язку, необхідної для перевірки точності набору тексту.

Salthouse допрацював і поліпшив запропоновану Купером модель процесу набору тексту. Butsch визначив, що «інтервал копіювання», тобто кількість тексту, що зберігається в пам'яті користувача при наборі, залежить від досвідченості і умінь користувача. Використання пам'яті в якості короткострокового буферу перед друкуванням було доведено експериментами, проведеними Thomas і Jones. Купер продемонстрував у своїх дослідженнях, що користувачі розбивають текст на маленькі передбачувані групи, через обмеження на розмір буферної пам'яті. Verwey і Dronkert в своїх експериментах довели, що процеси сприйняття читаного тексту і м'язових рухів при наборі тексту відбуваються одночасно. У дослідженні, проведеному Shaffer, було запропоновано існування внутрішнього регулярного ритму набору певних послідовностей символів. Він припустив, що клавіатурний почерк не є фіксованою характеристикою людини, а постійно змінюється. Він також зауважив, що процес набору заснований на знанні переходів і рухів між клавішами. Інтервали між послідовними натисканнями клавіші для досвідчених користувачів,

як показали спостереження, були меншими, ніж в користувачів, що мали кваліфікацію. R. S. Gaines, W. Lisowski, S. J. Press та N. Shapiro довели, що швидкість переміщення пальців у досвідчених машиністів вдвічі швидше, ніж у недосвідчених. Ostry в своєму дослідженні часу переміщень рук при друку показав, що організація руху пов'язана з поточним психофізіологічним станом людини.

На підставі проведеного аналізу перерахованих вище досліджень запропоновано вважати клавіатурний почерк індивідуальним та унікальним для кожної людини. Rumelhart і Norman провели комп'ютерне моделювання дій користувача, що працює з клавіатурою. Вони промоделювали час інтервалу між натискання клавіш і повторення помилок при введенні. Моделі Купера, Salthouse, Rumelhart і Norman є першими моделями, розробленими з метою аналізу властивостей і характеристик клавіатурного почерку користувача.

Під час набору тексту користувачем, можна відзначити три події: момент натискання клавіші, утримання, і момент відпускання клавіші. На основі вибірки моментів натискання і утримання клавіш розраховуються тимчасові характеристики почерку. Виділено наступні характеристики клавіатурного почерку, які застосовуються в системах ідентифікації і аутентифікації оператора:

- час інтервалу між натисканнями клавіш, тобто час від натискання однієї клавіші до моменту натискання наступної клавіші або час від моменту відпускання однієї клавіші до моменту натискання наступної. Використання цієї характеристики вимагає довгого збору статистичних даних. Розраховано, що для збору даних про час між натисканням 33 клавіш, відповідним букв російського алфавіту з розміром вибірки в 30 елементів потрібно 32670 елементів вхідних даних. Таким чином, дана характеристика почерку не може бути застосована для виявлення підміни законного користувача.
- Час введення біграмм (наприклад, «АА», «ОЕ», «ЯЯ»), триграм (наприклад, «ААА», «МАМ», «ЕЮЯ»), тобто час від натискання першої клавіші , до моменту відпускання останньої клавіші N-грами. Метод аналізу даної характеристики, так само як і попередній, вимагає збору великої кількості

статистичних даних. Цим обґрунтована непридатність даної характеристики для виявлення підміни користувача.

- Час утримання клавіш, тобто час від натискання до відпускання клавіші. Кількість вибірок статистичних даних часу утримання відповідає кількості натискає користувачем клавіш, тобто 33 букв російського алфавіту відповідає 33 вибірки часу утримання клавіш. Дана характеристика почерку дозволяє організувати процедуру постійного таємного клавіатурного моніторингу особистості користувача з метою виявлення підміни.
- Сила тиску, прикладеного до клавіші під час набору. Вимагає установки додаткових датчиків (по одному на кожну клавішу) в клавіатурі. Це призводить до збільшення складності розробки клавіатур, збільшення їх вартості і зростання ймовірності поломки клавіатури.
- Швидкість, тобто кількість набраних користувачем символів в відрізок часу, і ритм введення. Виявлено, що дана характеристика почерку залежить від психофізіологічного стану користувача і, тому, часто змінюється. Цим обґрунтована складність застосування даних характеристик в процесі виявлення підміни законного користувача.

У загальному вигляді, функція $\vartheta(t)$, що описує процес набору тексту користувачем на клавіатурі, може бути представлена наступним чином:

$$\vartheta(t) = \gamma(t)+\Theta(t)+\lambda(t) \text{ , де} \qquad (2.1)$$

$\gamma(t)$ – складова, що характеризує підсвідомі процеси мислення при наборі тексту;

$\Theta(t)$ – складова свідомих процесів мислення;

$\lambda(t)$ – механічні характеристики клавіатури, що впливають на процес набору тексту.

Основним завданням системи біометричної ідентифікації користувача за особливостями клавіатурного почерку є завдання виділення і подальшої

ідентифікації компоненту γ(t) з функції ϑ(t), яка визначає вихідні дані для системи ідентифікації після вимірювання тимчасових характеристик клавіатурного почерку користувача. Для цього необхідно виділити компоненти Θ(t) і λ(t) з вихідної функції ϑ(t). Зрозуміло, що через неможливість побудови механічної моделі рухів людини при наборі тексту єдиним прийнятним рішенням є збір статистичних даних по клавіатурному почерку великої кількості користувачів.

Імовірність аутентифікації користувача за часом утримання клавіш в залежності від довжини ключової фрази є значно більш стабільною характеристикою клавіатурного почерку оператора, ніж час між натисканнями клавіш, яке і зростає з ростом довжини ключової фрази. Це пояснюється тим, що процес натискання клавіші на клавіатурі є істинно підсвідомим процесом мислення. Характер даної функції практично не змінюється для широкого кола операторів незалежно від їх кваліфікації та досвіду роботи з клавіатурою. Звідси випливає, що складова γ(t) найбільш точно характеризується часом утримання клавіш при введенні тексту користувачем. Час утримання клавіші розраховується за формулою (2.2):

$$T_i^{утр.кл.} = T_i^{відп.кл.} - T_i^{натиск.кл.}, \text{ де} \qquad (2.2)$$

$T_i^{утр.кл.}$ – час утримання клавіші;

$T_i^{відп.кл.}$ – час відпускання клавіші;

$T_i^{натиск.кл.}$ – час натискання клавіші.

За підсумками аналізу характеристик клавіатурного почерку запропоновано використовувати час утримання клавіш в процесі виявлення підміни законного користувача. У шаблон почерку користувача запропоновано зберігати усереднені значення часу утримання клавіш. Маючи такий шаблон, стає можливо ідентифікувати користувача ключової системи. Далі наведено аналіз відомих методів ідентифікації і аутентифікації користувачів за клавіатурним почерком:

статистичних методів, методів заснованих на застосуванні нейронних мереж, розпізнаванні образів і генетичних алгоритмах.

Статистичні методи полягають в обчисленні відхилень характеристик поточного почерку користувача, який претендує на доступ до ІС і характеристик почерку в шаблоні, збереженого в системі для даного користувача. Для порівняння можуть використовуватися t-тести, Евклідова відстань, Хеммінгово відстань та інші. Joyce і Gupta використовуючи статистичні методи отримання точності аутентифікації користувача досягли FAR 0.25% і FRR 16.36%. У своїй роботі вони використовували Хеммінгово відстань (2.3) і відстань Махаланобіса (2.4):

$$M = \sum_{i=1}^{N} (|A_i| - |B_i|) \tag{2.3}$$

$$M = \sqrt{\sum_{i=1}^{N} \frac{(A_i - B_i)^2}{{\sigma_i}^2}} \tag{2.4}$$

i – номер клавіші;

N – кількість аналізованих клавіш;

A_i – тимчасова характеристика для клавіші з шаблону оператора, який претендує на доступ;

B_i – тимчасова характеристика для клавіші з шаблону, що зберігається в базі шаблонів і відповідного оператору;

σ – середньоквадратичне відхилення A_i від B_i.

Guven і Sogukpinar базуючись на векторному аналізі досягли точності ідентифікації оператора в 95%. Аналіз робіт зазначених дослідників дозволив зробити висновок, що головною проблемою статистичних методів є недолік даних на стадії навчання, тобто при отриманні шаблону почерку оператора.

Метод заснований на нейронних мережах вперше був застосований Obaidat і Macchiarolo для аутентифікації і ідентифікації користувачів за часом між натисканнями клавіш. Вони досягли точності ідентифікації оператора в 96.8%, використовуючи нейронну мережу, засновану на сумі творів. Yong та його колеги запропонували використовувати динамічну нейронну мережу. За аналізом даних досліджень зроблено висновок про те, що головна перевага нейронних мереж полягає в тому, що вони можуть обробляти відразу декілька параметрів почерку. Виділено основні недоліки застосування нейронних мереж при розпізнаванні клавіатурного почерку: потрібно чимало часу для процедур навчання та аутентифікації, присутність ситуацій, коли нейронна мережа не може навчиться через особливості вхідної вибірки. Так як даний метод використовується тільки в якості «чорного ящика», неможливо визначення достатнього для отримання шаблону почерку і подальшої успішної роботи обсягу і складу вхідної вибірки. Також, у разі додавання шаблону нового користувача до системи доведеться перенавчати всю нейронну мережу.

Для методу розпізнавання образів, Giot та його колеги запропонували використовувати метод опорних векторів для розпізнавання клавіатурного почерку. Вони досягли рівня ідентифікації в 95%. Виявлено головна перевага даного методу – висока точність, яка призведена ігноруванням помилок і похибок вимірювань часу утримання клавіш. Недоліком методу є те, що ідентифікація проводиться не за всіма шаблонами, а лише з тієї частини шаблонів, яка знаходиться на кордонах.

Revett та інші використовуючи генетичні алгоритми досягли ймовірності виникнення помилок FAR в 0.43% і FRR в 4.75%. Виявлено основна перевага використання генетичних алгоритмів – вони можуть легко взаємодіяти з великими базами даних та обробляти багатовимірні, недиференціальні, безперервні і непараметричні дані. Виділено головні недоліки генетичних алгоритмів: висока трудомісткість, що обмежує сферу застосування, низька стійкість до відхилень тимчасових характеристик почерку, даний метод не гарантує знаходження оптимального рішення.

2.2 Реалізація механізмів систем постійного клавіатурного моніторингу з метою виявлення підміни користувача

На основі зробленого аналізу пропонується наступна архітектура біометричної підсистеми виявлення підміни законного користувача (рисунок 1.3), що забезпечує процес отримання шаблону почерку і порівняння почерків.

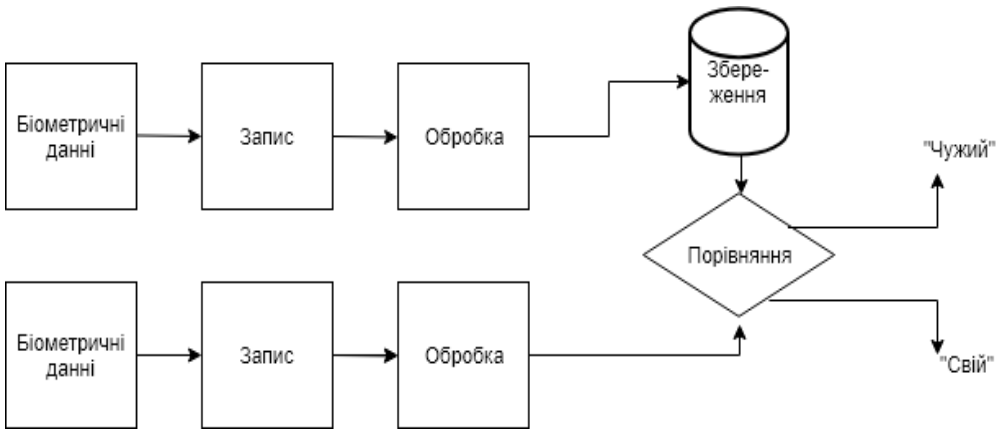


Рисунок 2.1 – Спрощена архітектура біометричної системи виявлення підміни законного оператора

Як видно на малюнку, підсистема складається з декількох блоків:

- Блок отримання і запису клавіатурного почерку користувача. Він відповідає за отримання тимчасових міток подій натискання і відпускання клавіш, та зазначає до якої клавіші відносяться ці події. Для отримання часу виникнення подій використовується таймер. Даний блок містить так само фільтр довгих натискань на клавіші. Важливим моментом є точність визначення моменту виникнення подій клавіатури. Отже, системи повинні бути зроблені стійкими до похибок вибірки часу.
- Блок обробки вибірки подій клавіатури відповідає за обчислення усереднених значень часу утримання клавіш.
- Блок зберігання клавіатурного почерку дозволяє зберегти шаблонні значення

клавіатурного почерку користувачів в базу шаблонів.

- Блок порівняння поточного клавіатурного почерку і шаблонного служить для порівняння почерків і прийняття рішення про аутентифікації і ідентифікації користувача за результатами порівняння.

2.3 Розроблення математичної моделі часу утримання клавіш

Аналіз клавіатурного почерку ґрунтуються на припущенні, що клавіатурний почерк представляється у вигляді усереднених значень подій клавіатури. У системах Microsoft Windows виділяють три види подій клавіатури:

- Подія KeyDown, яка відбувається один раз. Спрацьовує під час натискання фізичної клавіші. Подія нижчого рівня – реагує на натискання будь-якої клавіші на клавіатурі. Повертає код натиснутою клавіші.
- Подія KeyUp, яка виникає один раз після того, як користувач відпускає фізичну клавішу. В усьому іншому подія аналогічна до KeyDown.
- Подія KeyPress, яка може виникати кілька разів, коли користувач утримує натиснуту клавішу. Ця подія виникає при натисканні клавіші, що призводить до введення знаку.

У системах розпізнавання клавіатурного почерку статистичними даними є значення часу подій клавіатури. Обраною ознакою клавіатурного почерку є час утримання клавіш, яке відповідає часовому інтервалу між подіями KeyDown(A) і KeyUp(A), де A – одна з клавіш клавіатури. У зв'язку з використанням даного методу необхідний збір статистики, що складається з вибірки тимчасових значень, де елементом вибірки буде час утримання клавіші.

У ймовірно-статистичному формулюванні виникає необхідність побудови середньостатистичних шаблонів на основі зразків, пред'явлених системі в режимі навчання. При цьому слід враховувати, що на характеристики клавіатурного почерку людини впливає безліч факторів: програмні і апаратні затримки, які теж є випадковими величинами, рух нервового імпульсу по нейронам, час відгуку м'язів

людини на сигнал посланий мозком та інші. Значить, на клавіатурний почерк впливає безліч незалежних випадкових величин. Ефект їх складання описується формулою Гауса. Відповідно для зменшення впливу випадкових помилок необхідно провести вимірювання досліджуваної величини кілька разів.

Розглянемо застосування формули Гауса в процесі обробки результатів вимірювання характеристик клавіатурного почерку. Припустимо, що ми вимірюємо час утримання якоїсь конкретної клавіші, позначимо цю величину X. В результаті проведених вимірювань ми отримали вибірку значень величини (2.5):

$$X_1,\ X_2,\ X_3 \ldots X_N \qquad (2.5)$$

Цей ряд значень величини X складе нашу вибірку часу утримання клавіші. По даній вибірці дається оцінка результату вимірювань, тобто усереднене значення ЧУК, в яке прагне укластися користувач при натисканні на клавішу. Величину, яка буде такою оцінкою, ми позначимо $\bar{X}$. Але так як це значення оцінки результатів вимірів не буде являти собою істинного значення вимірюваної величини часу утримання клавіші, необхідно оцінити помилку вимірювання. Припустимо, що ми зуміємо визначити оцінку помилки ΔX. У такому випадку ми можемо записати результат вимірювань у вигляді (2.8):

$$\mu = \bar{X} \pm \Delta X\,. \qquad (2.6)$$

Таким чином, маючи вибірку, необхідно знайти оцінку результату вимірювань $\bar{X}$, його помилку ΔX і надійність P. Це завдання вирішується застосуванням математичної статистики. Запропоновано в якості оцінки результатів вимірювань ЧУК розраховувати середнє значення всіх елементів зібраної для конкретної клавіші вибірки (2.7):

$$\overline{X} = \frac{\sum_{i=1}^{N} X_i}{N} \tag{2.7}$$

Де N – число вимірювань.

Значить, для вибірки в N вимірювань часу утримання, найбільш імовірним значенням вимірюваної величини буде її середнє арифметичне значення. Отримане середнє значення ЧУК прагне до істинного значення μ вимірюваної величини $\bar{X}$ при збільшенні числа вимірювань, тобто $N \to \infty$.

Середньоквадратичної помилкою середнього арифметичного називається величина:

$$S_{\overline{X}} = \sqrt{\frac{\sum(\overline{X} - X_i)^2}{N(N-1)}} = \frac{S}{\sqrt{N}} \tag{2.8}$$

Точність оцінки зростає при збільшенні числа вимірювань. Помилка $S_{\overline{X}}$ дозволяє оцінити точність, з якою розраховано середнє значення ЧУК.

Отже, можна визначити, скільки разів повинна бути натиснута конкретна клавіша, для того щоб зібрати статистику, що характеризує усереднений ЧУК. Відповідно, для того щоб визначити клавіатурний почерк користувача, потрібно зібрати дані про середні значення часу утримання всіх використовуваних клавіш.

2.4 Розроблення аналітичної моделі клавіатурного почерку

Маючи шаблон клавіатурного почерку користувача стає можливо провести його аутентифікацію та ідентифікацію. Для цього необхідно провести процедуру порівняння поточного зразка почерку і збереженого раніше шаблона. Запропоновано використовувати Евклідова відстань для порівняння часу утримання клавіш поточного зразка і шаблону (2.9):

$$M = \sqrt{\sum_{i=1}^{V} (A_i - B_i)^2}\text{, де} \qquad (2.9)$$

M – розраховане значення Евклідова відстані,

V – кількість вибірок часу утримання клавіші, що відповідає кількості аналізованих клавіш,

A_i – час утримання клавіші з поточного зразка клавіатурного почерку користувача, який претендує на доступ,

B_i – час утримання клавіші, що зберігається в шаблоні почерку.

Користувач буде успішно ідентифікований або його особистість буде підтверджена, якщо розраховані значення Евклідова відстані менше встановленого в системі порога доступу. Поріг доступу підбирається в залежності від вимог до розроблюваної системи. Основними вимогами для систем захисту інформації є ймовірності виникнення помилок першого і другого роду.

2.5 Розроблення методу розпізнавання клавіатурного почерку користувача

Запропоновано метод розпізнавання клавіатурного почерку за часом утримання клавіш та інтервалом між натисканням клавіш. Це дозволить отримати шаблон почерку, який не залежить від тексту, що набирається і порядку вводу символі. Таким чином стає можливо визначення клавіатурного почерку користувача ключової системи по вільному контрольному тексту. Це забезпечує можливість застосування методу для задач постійного прихованого клавіатурного моніторингу з метою виявлення підміни авторизованого законного користувача, визначення відхилення психофізіологічного стану користувача ключової системи від нормального.

3 ДОСЛІДЖЕННЯ ОСНОВНИХ ПІДХОДІВ ДО ІДЕНТИФІКАЦІЇ КОРИСТУВАЧІВ НА ОСНОВІ КЛАВІАТУРНОГО ПОЧЕРКУ

3.1 Огляд параметрів використовуваних в дослідженні

Основним параметром, який використовується при аналізі клавіатурного почерку, є час між подіями клавіш клавіатури. Ідентифікація користувача будується на аналізі тривалості інтервалів, відповідних кожної комбінації клавіш (рис. 1.)

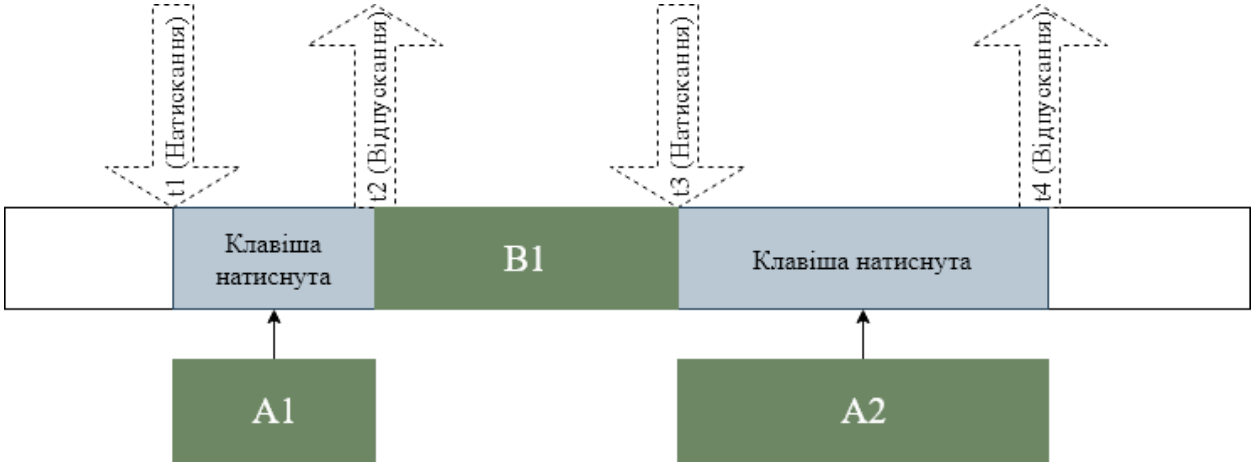


Рисунок 3.1 – Часова діаграма клавіатурного почерку

Для реалізації схеми зміни клавіатурного почерку пропонується використання програмного або апаратного модуля. Можливі такі основні місця вбудовування такого модуля:

- безпосередньо в клавіатуру (програмно-апаратний або апаратний модуль);
- безпосередньо в обробник подій клавіатури операційної системи (програмний модуль);
- між клавіатурою і комп'ютером (апаратний модуль в розрив кабелю). Розглянемо кожен з варіантів більш докладно.

Вбудовування модуля безпосередньо в клавіатуру вимагає внесення змін у конструкцію клавіатури і, можливо, в її драйвер. Будь-яких принципових переваг, крім візуальної скритності, таке розміщення модуля не несе.

Програмний модуль обробки подій клавіатури не вимагає яких-небудь апаратних компонентів, проте його функціонування залежить від операційної системи. По суті, кожна операційна система вимагає свого програмного модуля. Такий модуль не залежить від того, через який порт підключається клавіатура.

Установка модуля зміни клавіатурного почерку в розрив кабелю дозволяє використовувати його на будь-який ОС. Недоліком же такого рішення є залежність від порту підключення (найбільш часто використовуються PS / 2 або USB). В даній роботі розглянемо програмний модуль.

3.2 Дослідження та аналіз отриманих значень

Для перевірки були використані зразки клавіатурного почерку п’яти людей. Час утримання клавіши у першого користувача для першого зразка варіюється від 85 мс до 114 мс, час між натисканнями клавіш – від 92 мс до 481 мс. При цьому середнє утримання клавіш 98,6 мс, а середній час між натисканням клавіш складає 208,5 мс (рисунок 3.2).

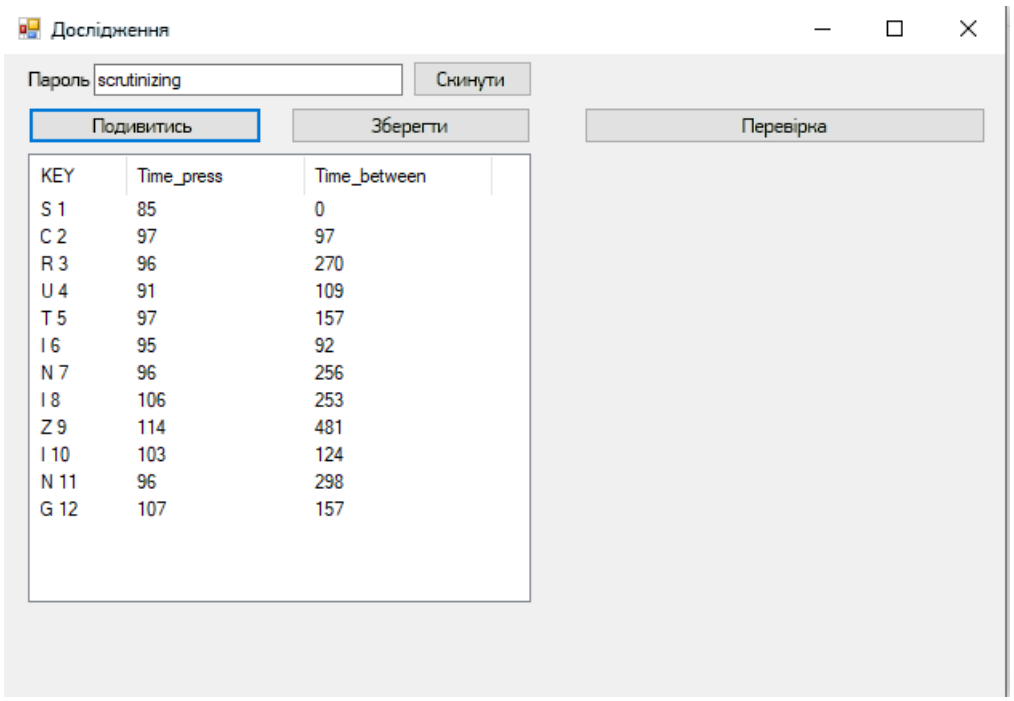


Рисунок 3.2 – Результат КП для першого зразка першого користувача

Час утримання клавіши у першого користувача для другого зразка варіюється від 75 мс до 112 мс, час між натисканнями клавіш – від 28 мс до 143 мс. При цьому середнє утримання клавіш 88,3 мс, а середній час між натисканням клавіш складає 93,6 мс, тобто менше, аніж в першому зразку (рисунок 3.3).

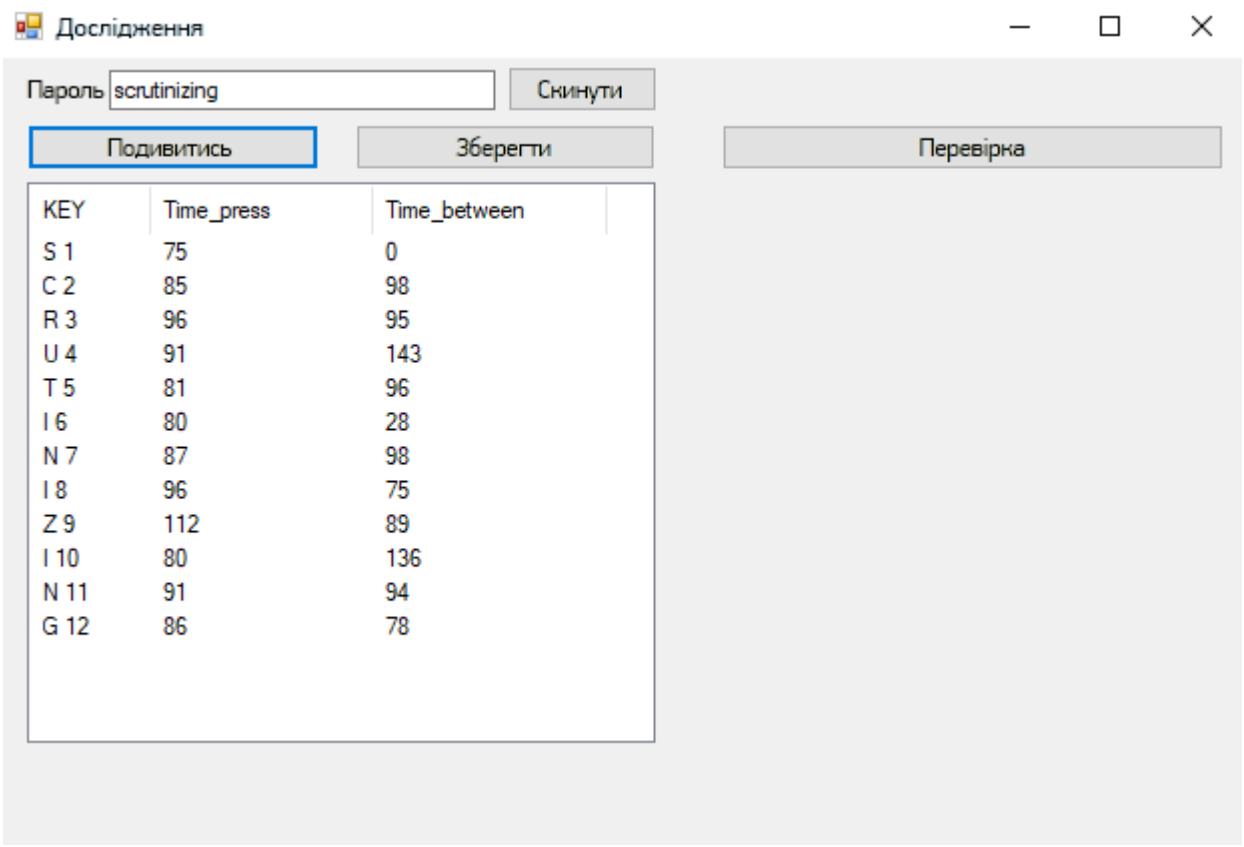


Рисунок 3.3 – Результат КП для другого зразка першого користувача

Час утримання клавіши у другого користувача варіюється від 75 мс до 107 мс, час між натисканнями клавіш – від 31 мс до 234 мс. При цьому середнє утримання клавіш 91 мс, а середній час між натисканням клавіш складає 91,8 мс (рисунок 3.4).

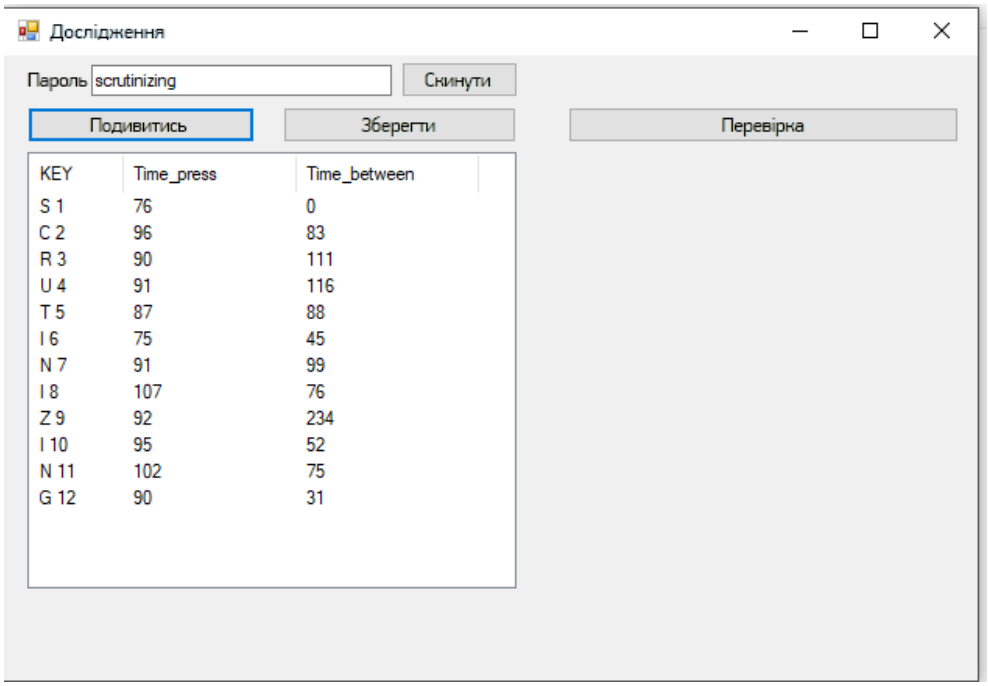


Рисунок 3.4 – Результат КП другого користувача

Час утримання клавіши у третього користувача варіюється від 95 мс до 160 мс, час між натисканнями клавіш – від 55 мс до 736 мс. При цьому середнє утримання клавіш 111,3 мс, а середній час між натисканням клавіш складає 327 мс (рисунок 3.5).

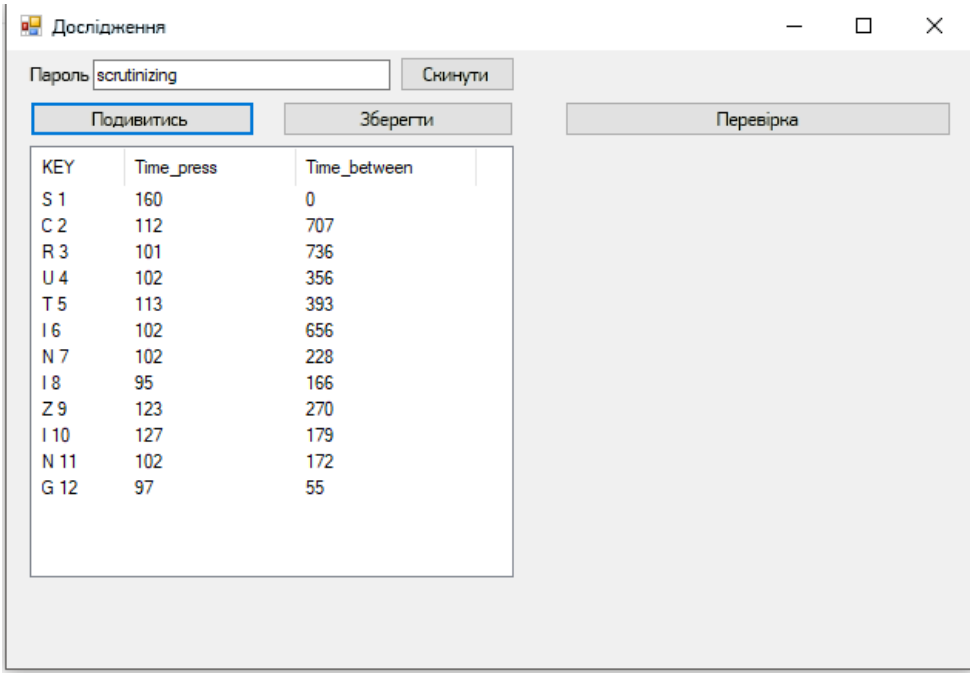


Рисунок 3.5 – Результат КП третього користувача

Час утримання клавіши у четвертого користувача варіюється від 91 мс до 128 мс, час між натисканнями клавіш – від 145 мс до 1287 мс. При цьому середнє утримання клавіш 104,8 мс, а середній час між натисканням клавіш складає 615,5 мс (рисунок 3.6).

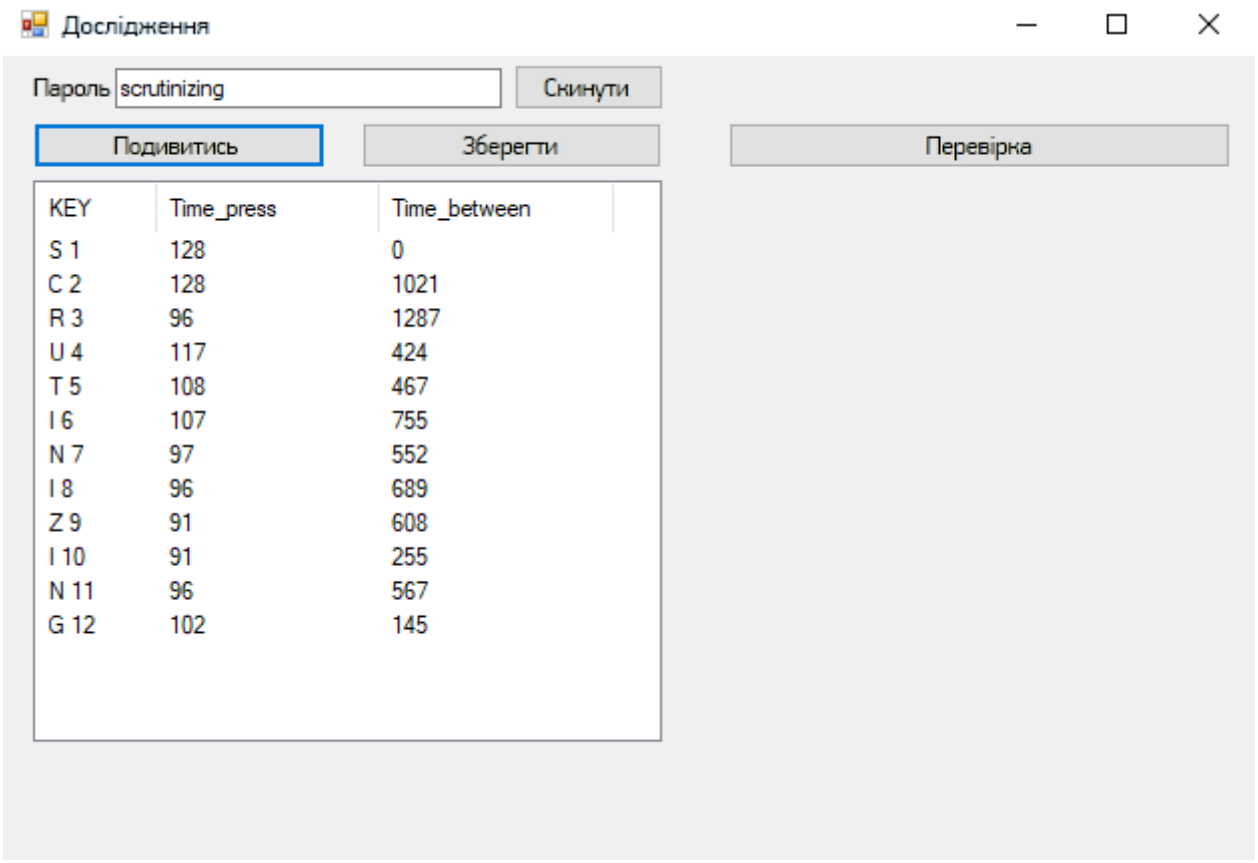


Рисунок 3.6 – Результат КП четвертого користувача

Час утримання клавіши у п’ятого користувача варіюється від 68 мс до 102 мс, час між натисканнями клавіш – від 51 мс до 507 мс. При цьому середнє утримання клавіш 82 мс, а середній час між натисканням клавіш складає 233,2 мс (рисунок 3.7).

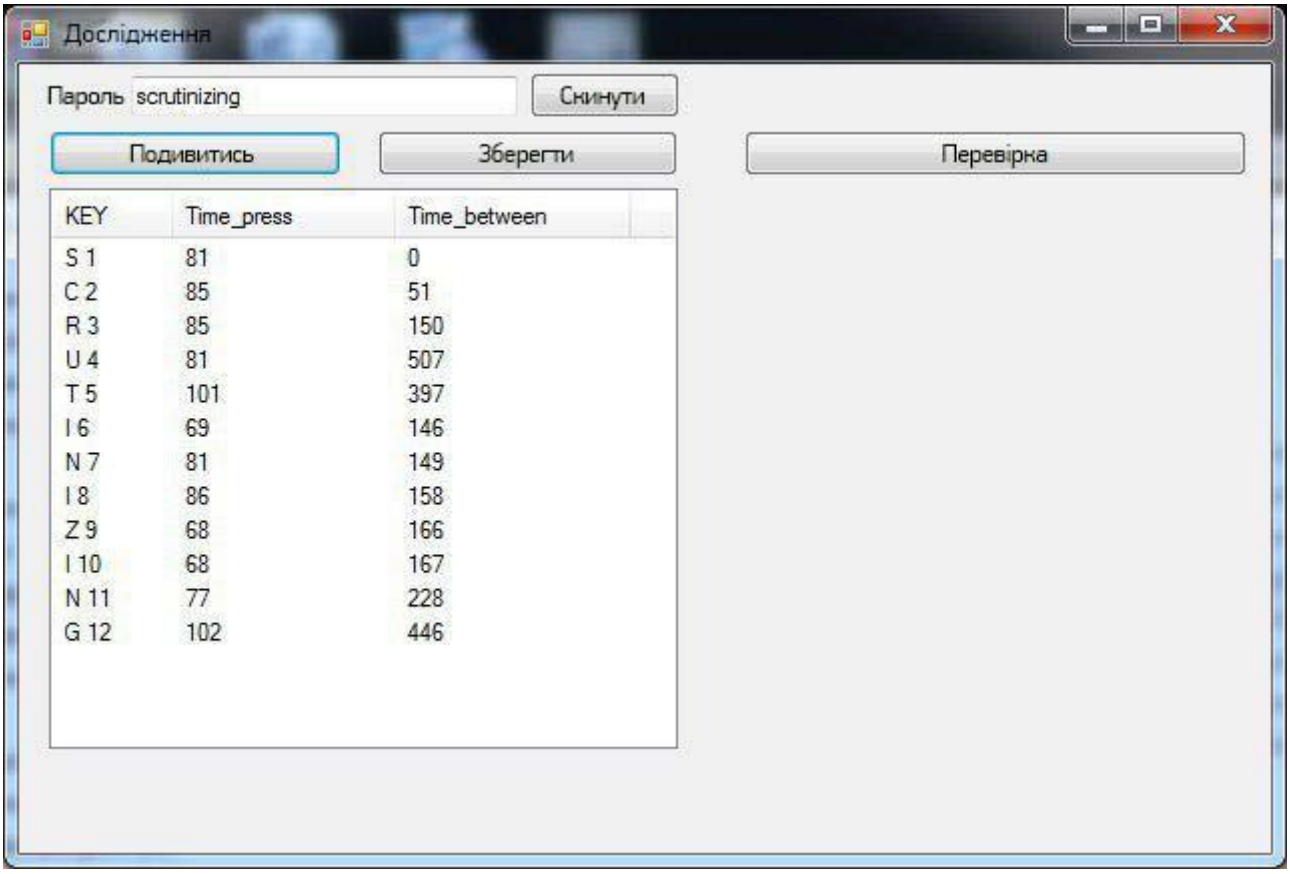


Рисунок 3.7 – Результат КП п’ятого користувача

Після закінчення тестування з отриманих даних було побудовано графіки:

- порівнянь часу утримання клавіш двох зразків першого користувача (рисунок 3.8);
- порівняння часу між натисканням клавіш двох зразків у першого користувача (рисунок 3.9);
- порівняння часу утримання клавіш двох користувачів (рисунок 3.10);
- порівняння часу між натисканням клавіш у двох користувачів (рисунок 3.11);
- порівняння часу утримання клавіши у п’яти користувачів, порівняння часу між натисканням клавіш у п’яти користувачів зображено на рисунках 3.12 та 3.13 і винесено в додаток К.

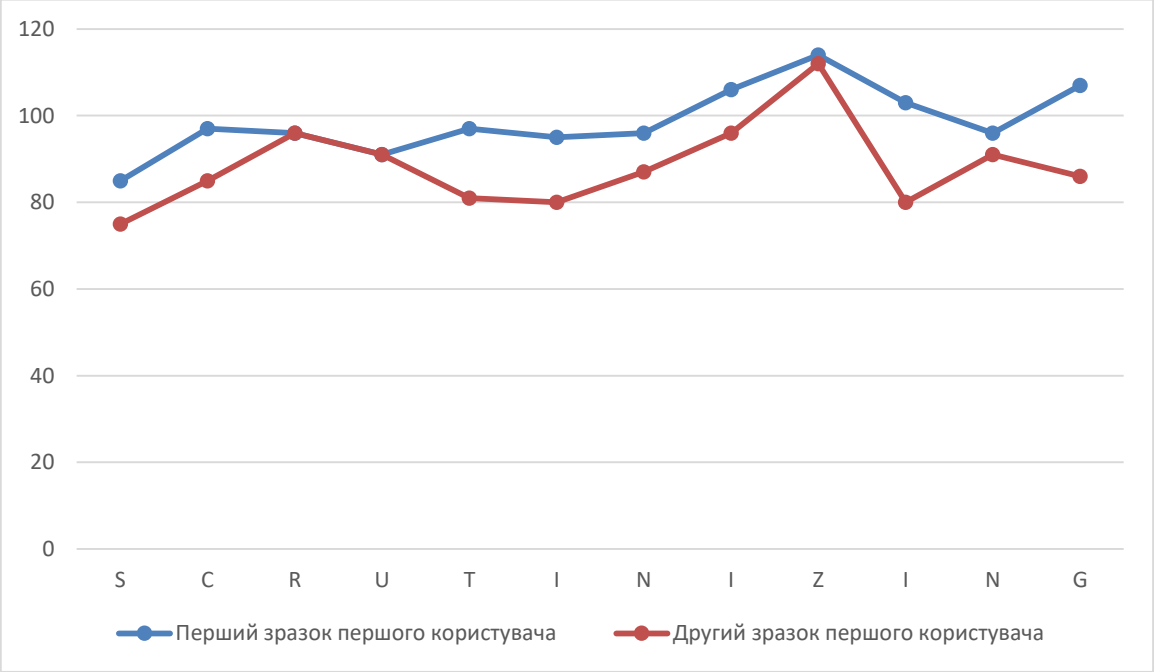


Рисунок 3.8 – Порівняння часу утримання клавіш двох зразків у першого користувача

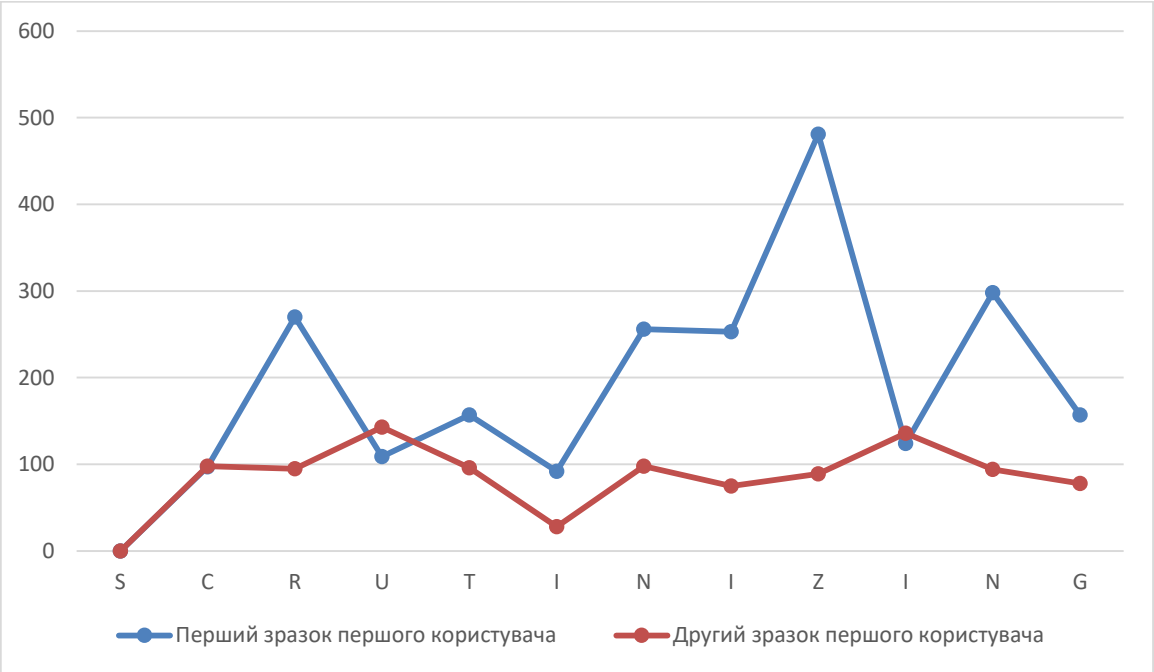


Рисунок 3.9 – Порівняння часу між натисканням клавіш двох зразків у першого користувача

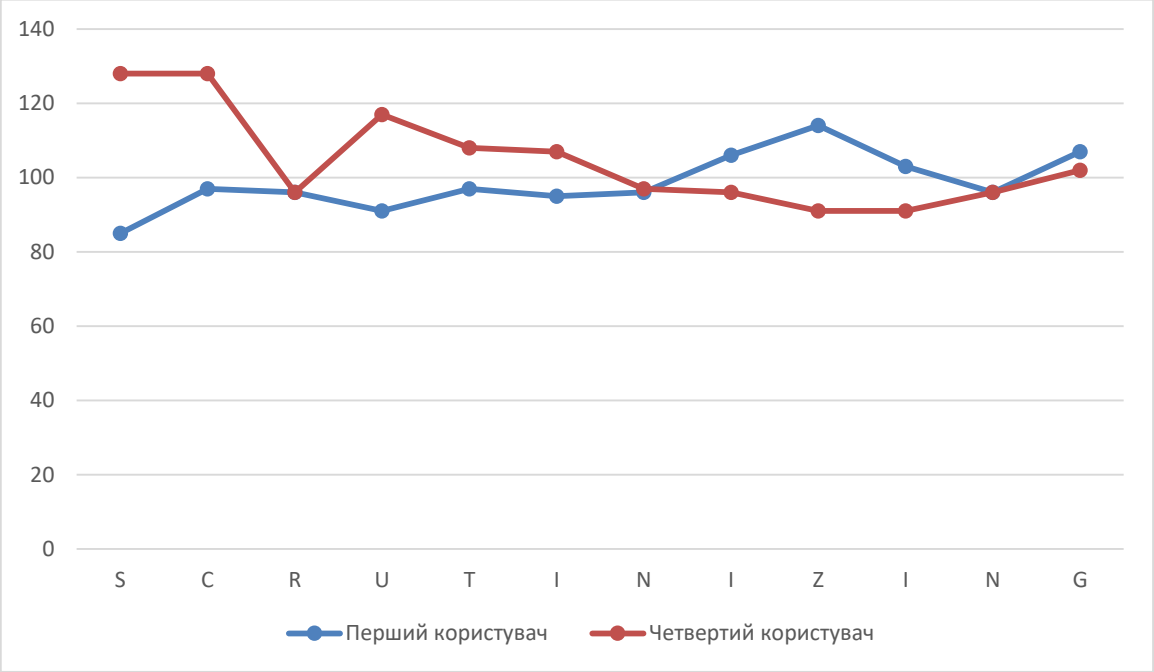


Рисунок 3.10 – Порівняння часу утримання клавіш двох користувачів

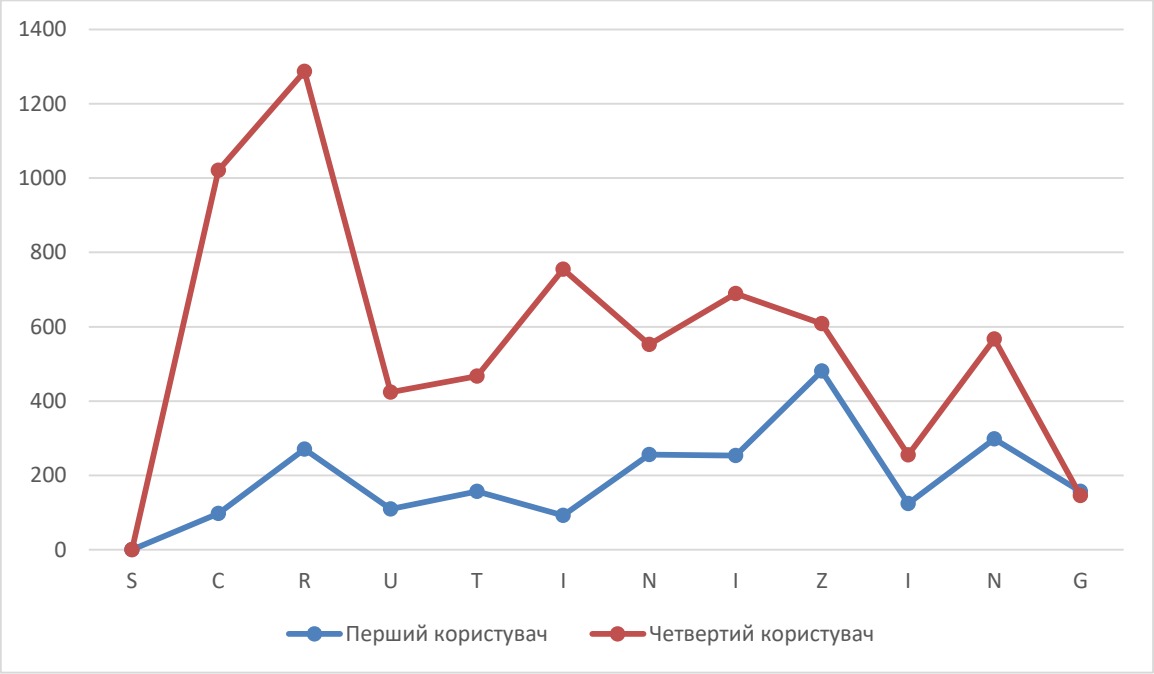


Рисунок 3.11 – Порівняння часу між натисканням клавіш у двох користувачів

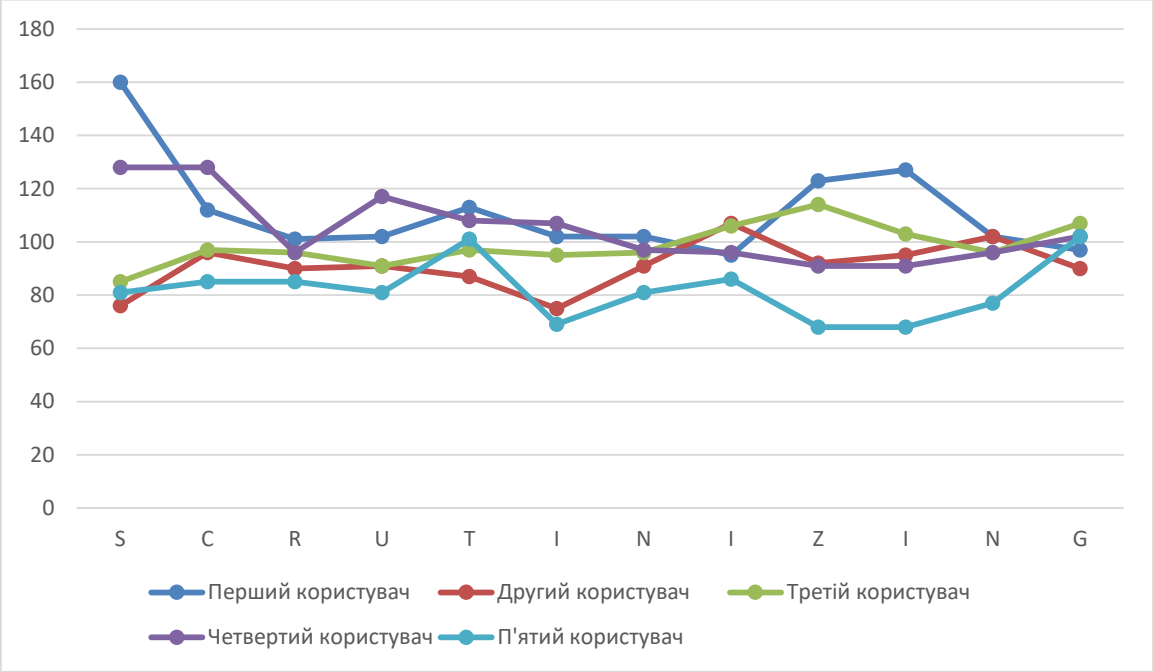


Рисунок 3.12 – Порівняння часу утримання клавіши у п’яти користувачів

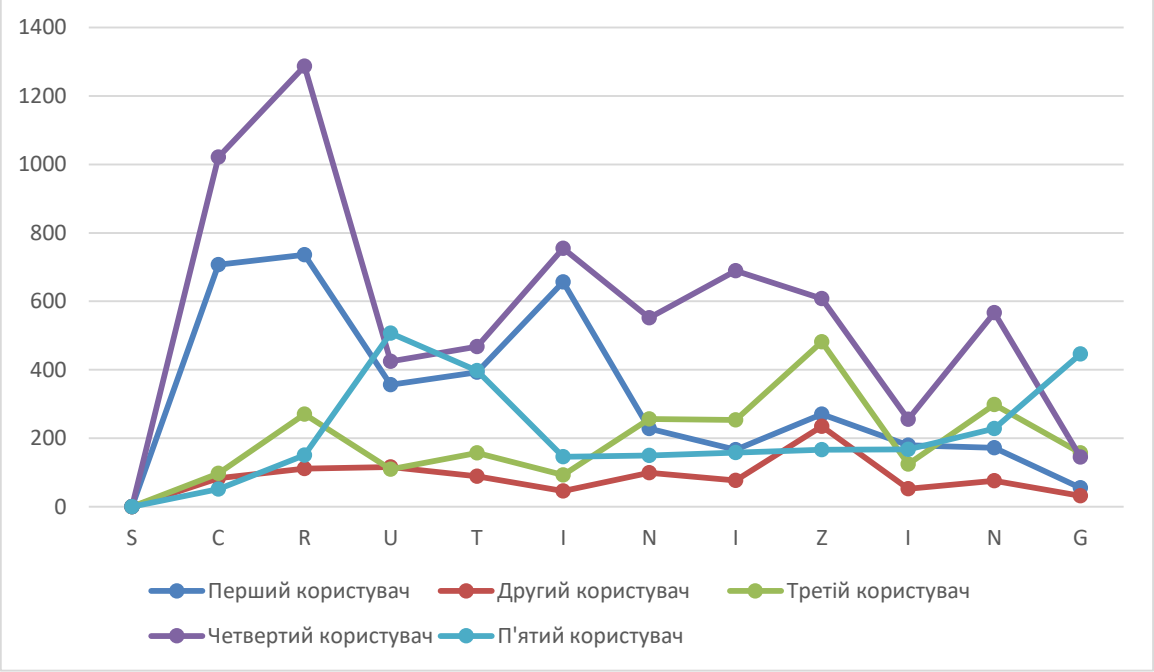


Рисунок 3.13 – Порівняння часу між натисканням клавіш у п’яти користувачів

Таким чином, отримані в ході тестування данні дають змогу зробити висновок, що кожній людині притаманний свій клавіатурний почерк, а почерки різних людей відрізняються між собою. Виходячи з цього, аутентифікацію на основі аналізу клавіатурного почерку має свою ефективність та може бути використана.

4 ПРАКТИЧНА РЕАЛІЗАЦІЯ СИСТЕМИ ІДЕНТИФІКАЦІЇ НА ОСНОВІ КЛАВІАТУРНОГО ПОЧЕРКУ ТА ДОСЛІДЖЕННЯ ЇЇ ЕФЕКТИВНОСТІ

За основу розробки було взято платформу My Visual Database. Ця платформа досить проста та широко використовується серед розробників для створення простих та складних баз даних. Створена програма в даній платформі являється повноцінним windows додатком, яке не потребує встановлення додаткових, сторонніх компонентів та додаткового програмного забезпечення, адже за замовчуванням використовує базу даних SQLite, що не потребує додаткових налаштувань. Також використовується бібліотека FastScript для виконання скриптів. Вона дає змогу додати можливості виконання скриптових програм до проекту. Унікальністю FastScript являється можливість одночасного використання декількох мов програмування. FastScript не використовує Microsoft Scripting Host, а тому може використовуватися як в Windows, так і в Linux.

В програмі протягом всього часу її роботи використовується принцип неперервного вимірювання таймінгів. Отримання же шаблону, тобто середніх величин, який використовується для порівняння з раніше записаними зразками під час реєстрації користувача, здійснюється з початком виконання спеціальних дій, таких як закінчення вводу пароля та закінчення вводу тексту. Під час формування зразкового шаблону використовується ще подія закінчення введення серії повторів згенерованого пароля. Всі важливі події, що здійснюються в програмі, фіксуються в базі даних у вигляді журналу подій.

4.1 Сценарій використання системи

UML діаграму сценаріїв використання системи ідентифікації на основі клавіатурного почерку з описом кожного прецеденту наведено нижче на рисунку в додатку А.

4.2 Запис даних в кільцевий стек

Для запису таймінгів використовуються події натиснення клавіш і відпускання клавіш. В програмі для цих цілей є два обробники:

1. Sign_OnKeyDown (); — створений для реєстрації натиснення клавіші.
2. Sign_OnKeyUp (); — створений для реєстрації відпускання клавіші.

В додатку Б процеси зображені як один.

Кільцеві стеки реалізовані як двовимірні масиви KeyPress:

1. array [1..MAX_KEY_CODE, 1..MAX_SER_LEN] of integer; — масив, в якому зберігається час утримання клавіші.
2. KeyInt: array [1..MAX_KEY_CODE, 1..MAX_SER_LEN] of integer; — масив, в якому зберігається час перед натисканням клавіші, тобто інтервал між натисканням.
3. KeyDown: array [1..MAX_KEY_CODE] of TTime; — масив, у котрому зберігається момент натиснення клавіші.
4. KeyIndex: array [1..MAX_KEY_CODE] of integer; — даний масив використовується як індекс для зберігання серії даних і організації кільцевого стека зберігання.
5. KeyCount: array [1..MAX_KEY_CODE] of integer; — останній масив, у якому зберігаються лічильники натискань клавіш.

Вимірювання тривалості натискання клавіш відбувається наступним чином: під час натиснення клавіші в масиві KeyDown [] зберігається проміжок часу, що був витрачений на натискання цієї кнопки. При відпуску ж клавіші збережене значення використовується для визначення часу утримання клавіші в мілісекундах, як різниця між часом натискання та часом утримання. Час утримання зберігається в двовимірний масив KeyPress [], в якому перший індекс відповідає коду клавіші, а другий — вказівнику запису кільцевого стека, який вже зберігається в масиві KeyIndex []. Таким чином, для кожної клавіші є можливість збереження декількох

виміряних значень часу утримання. Кількість фактично проведених вимірювань зберігається в масиві KeyCount []. У випадку, якщо кількість вимірювань перевищить максимальну місткість стека, яка визначається константою MAX_SER_LEN, то покажчик стека переміститься на початок масиву.

Для вимірювання інтервалів між натисканнями використовується час попереднього натиснення клавіші, яке зберігається в глобальній змінній PredPress: TTime; і час поточного натиснення. Результат зберігається в двовимірний масив KeyInt [], запис в якій виконується аналогічно запису в масив KeyPress[]. Для того, щоб занадто великі паузи в наборі даних, які можуть бути викликані якимись зовнішніми подіями, такими як телефонний дзвінок чи повідомлення, та не вносили великих спотворень в вимірювання таймінгів, перед записом значення проводиться його порівняння з пороговим, і, в випадку перевищення, записується фіксоване значення 50 мс.

4.3 Отримання шаблону

При створенні однієї з спеціальних подій натиснення клавіш або відпускання клавіш, викликається процедура CalcStatistic; в якій проводяться обчислення середніх величин таймінгів. Для цього перебирається масив KeyCount [] для всіх клавіш, але для обчислення використовуються тільки ті елементи, які відповідають натисненні тих клавіш, які були використані користувачем в поточному сеансі. Для підрахунку середніх арифметичних значень часу утримання кожної клавіші і часу перед натисканням кожної клавіші, тобто пошуку її, використовується найменша величина з двох – або лічильник числа натискань KeyCount [], або розмір кільцевого стека. Результат зберігається в масиви KeyPressOvr [] та KeyIntOvr [], після чого обчислюються ще дві величини: середній час утримання (KeyPressTOvr – характеризує силу натискання на клавіші) і середній час між натисненням (KeyIntTOvr – характеризує темп набору тексту), які також беруть участь в

ідентифікації користувача. Сам алгоритм отримання шаблону зображений на рисунку 4.1 та в додатку В.

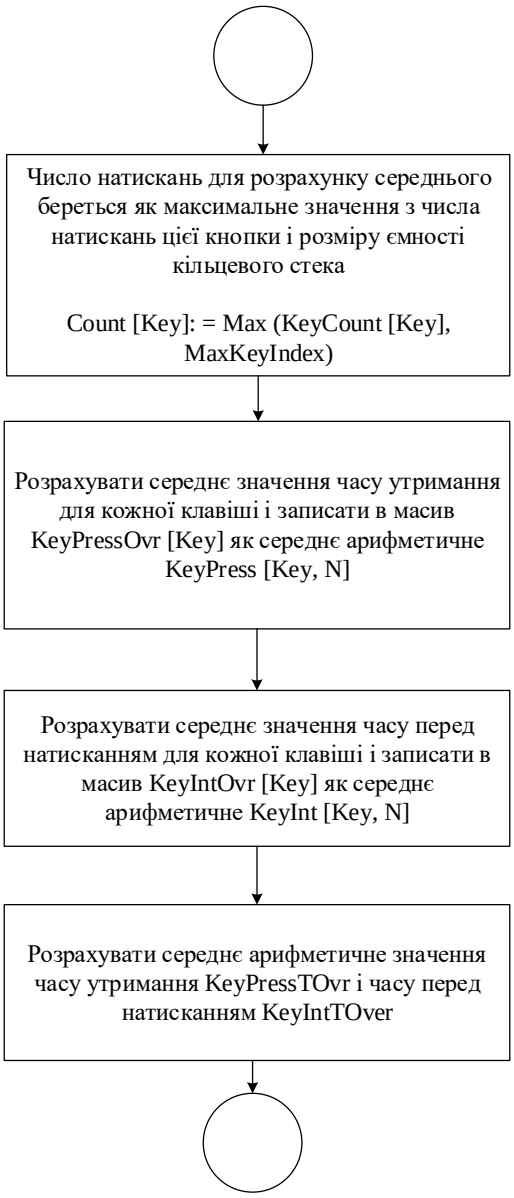


Рисунок 4.1 — Отримання шаблону

Збереження шаблону в базу даних здійснюється процедурою SaveTiming (), в якості параметру передається тип шаблону. Розрізняються два типи шаблонів: пароля та набору тексту. Це зроблено тому, що склад натиснутих клавіш в режимі введення пароля і в режимі набору тексту істотно різниться, і, якщо використовувати для ідентифікації тільки один тип шаблону на обидва процеси, збіжність буде дуже низькою.

Дані зберігаються в таблиці "timing" і "user". Щоб завантажити таймінг для поточного користувача використовується процедура LoadTiming ();

4.4 Алгоритм авторизації

Авторизація – це регулювання ступенями доступу до певного захищеного ресурсу та надання користувачеві прав для входження до системи. Вона проводиться через реєстрацію пароля користувача, а також запис шаблонів таймінгів набору пароля і набору контрольного тексту.

Під час першого входу користувача в систему, програма пропонує йому пройти авторизацію. Для цього він повинен натиснути кнопку для генерації пароля. Алгоритм створення випадкового паролю реалізований в обробнику події frmChangePass_brnCreatePass_OnClick (); Поки користувач набирає пароль задану кількість разів, проводиться запис таймінгів в кільцевий стек, а після завершення введення виконується отримання шаблону і запис його в базу даних. Аналогічно проводиться другий тип вимірювань для отримання шалону таймінгів при наборі випадкового тексту. Даний алгоритм зображений в додатку Г.

4.5 Алгоритм аутентифікації

Аутентифікація – це процес розпізнавання користувача. Він проводиться на основі перевірки правильності вводу пароля а також порівнянні шаблонів таймінгів користувача з збереженими шаблонами в базі раніше.

Під час набору користувачем пароля проводиться запис таймінгів в кільцевий стек. Після закінчення набору відбувається отримання шаблону по таймінгам з кільцевого стеку. Після перевірки пароля цей шаблон порівнюється зі збереженим в базі шаблоном користувача. На підставі порівняння виноситься вердикт, чи пройшов користувач аутентифікацію чи ні. Алгоритм аутентифікації зображений в додатку Д.

Під час порівняння шаблонів визначається максимальне відхилення між усіма порівнюваними параметрами, такими як; час утримання кожної клавіші, пауза перед натисканням кожної клавіші, середнім часом утримання і середньою паузою перед натисканням. Якщо це відхилення перевищує порогове, величина якого зберігається в базі даних, то порівняння вважається проваленим.

4.6 Алгоритм моніторингу

Моніторинг – це аутентифікація, що приховано виконується в процесі роботи користувача, після того, як він увійшов в систему і приступив до роботи.

Прихований моніторинг реалізований аналогічно до аутентифікації, але обробники натискань і відпускання клавіш підключені до поля, в якому користувач набирає робочий текст. Час моніторингу – це весь період часу, що був витрачений на введення тексту. Весь час, що користувач працює в програмі – ведеться запис таймінгів в кільцевий стек. Після того, як користувач набрав документ і зберіг його в базі даних, відбувається отримання шаблону і порівняння його з шаблоном користувача, збереженим в базі. Якщо порівняння провалено, то робота програми завершується. Алгоритм моніторингу зображений в додатку Е.

4.7 Схема бази даних

На схемі баз даних, зображеній на рисунку 4.2 та додатку Ж, представлені таблиці, в яких зберігаються шаблони таймінгів, а також дані по користувачам, налаштування програми та інша інформація.

Таблиця 4.1 — Опис позначок полів бази даних

	Ключове поле. У цьому полі зберігається унікальний ідентифікатор запису (UID).
	Зовнішній ключ. Дане поле пов'язане з ключовим полем в іншій таблиці (Foreing key).

●	Обов'язкове поле. Дане поле обов'язково має містити значення відмінне від NULL
○	Необов'язкове поле.

Таблиця 4.2 — Опис таблиць бази даних

user	Дані про користувачів: 1. id – ключове поле; 2. login – логін користувача; 3. password – пароль користувача; 4. is_admin – ознака того, що користувач являється адміністратором; 5. is_getident – ознака проходження авторизації; 6. pretime – середній час між натисканням клавіш; 7. holdtime – середній час утримання клавіш;
timing	Шаблон таймінгів: 1. id – ключове поле; 2. key – код клавіши; 3. pretime – середній час між натисканням клавіші; 4. holdtime – середній час утримання клавіші; 5. id_user – користувач; 6. type – тип шаблону 0 – для пароля, 1 – для тексту;
param	Параметри програми: 1. id – ключове поле; 2. k_dev – коефіцієнт максимального відхилення при порівнянні шаблонів; 3. ValidPassKey – допустимі символи для пароля; 4. PassEnterCount – кількість повторень пароля при проведенні

	авторизації;
data	Дані, що використовуються (набираються) користувачами - щоденники спостережень: 1. id – ключове поле; 2. id_user – користувач; 3. create_date – дата створення запису; 4. caption – заголовок запису; 5. data – вміст запису;
log	Журнал подій: 1. id – ключове поле; 2. event_time – дата події; 3. event_desc – опис події;
testdata	Тестові дані для отримання шаблонів таймінгів набору тексту 1. id – ключове поле; 2. data – текст для набору; 3. order_num – порядковий номер;

Рисунок 4.2 — Схема бази даних

4.8 Огляд та пояснення основних частин коду програми

Напочатку файлу знаходиться розділ Const (рисунок 4.3) – це розділ, який описує константи:

- обсяг статистики натиснень кожної клавіши, що зберігається;
- максимальне значення коду;
- тип таймінгу введення логіну;
- тип таймінгу введення тексту;
- мінімальну довжину пароля;
- можливі коливання довжини пароля.

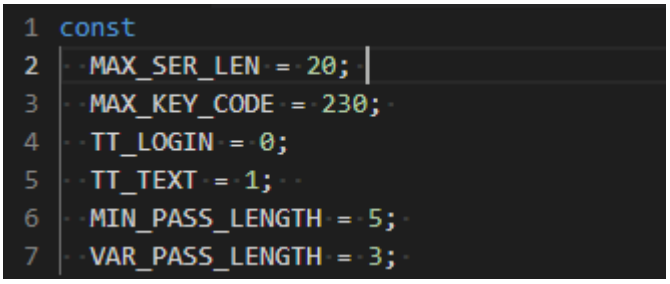


Рисунок 4.3 — Розділ з описаними константами

Він повинен знаходитися перед розділом Var, що зображений на рисунку 4.4. Цей розділ використовується для позначення секції, де оголошуються змінні і їх типи. Змінні зазвичай оголошуються на початку програми, процедури, функції або модуля.

```
var
  ValidPassKey: string;
  EnterPassCount: integer;
  UserID: integer;
  PredPress: TTime;
  KeyDown : array[1..MAX_KEY_CODE] of TTime;
  KeyIndex: array[1..MAX_KEY_CODE] of integer;
  KeyCount: array[1..MAX_KEY_CODE] of integer;
  KeyPress: array[1..MAX_KEY_CODE,1..MAX_SER_LEN] of integer;
  KeyInt: array[1..MAX_KEY_CODE,1..MAX_SER_LEN] of integer;
  KeyPressOvr: array[1..MAX_KEY_CODE] of integer;
  KeyIntOvr: array[1..MAX_KEY_CODE] of integer;
  KeyPressTOvr: integer;
  KeyIntTOvr: integer;
  UserKeyPressOvr: array[1..MAX_KEY_CODE] of integer;
  UserKeyIntOvr: array[1..MAX_KEY_CODE] of integer;
  UserKeyPressTOvr: integer;
  UserKeyIntTOvr: integer;

procedure frmTiming_OnShow (Sender: TObject; Action: string);
```

Рисунок 4.4 — Секція з оголошенням змінних

Нижче, на рисунку 4.5, зображено підключення обліку таймінгів для зняття сигнатур натискання клавіші, та на рисунку 4.6 для зняття сигнатур відпускання клавіші.

```
82 procedure Sign_OnKeyDown (Key: Word);
83 var
84   tmpPressTime:TTime;
85   tmpInterval: integer;
86 begin
87   if (Key >= 1) and (Key <= MAX_KEY_CODE) then
88   begin
89     tmpPressTime := Time();
90     KeyDown[Key] := tmpPressTime;
91     inc( KeyCount[Key] );
92     try
93       tmpInterval := MilliSecondsBetween(tmpPressTime,PredPress);
94     except
95       tmpInterval := 50;
96     end;
97     KeyInt[Key,KeyIndex[Key]] := tmpInterval;
98     PredPress := tmpPressTime;
99   end;
100 end;
```

Рисунок 4.5 — Облік таймінгу зняття сигнатур натискання клавіші

```
105 procedure Sign_OnKeyUp (Key: Word);
106 var
107   tmpInterval: integer;
108 begin
109   if (Key >= 1) and (Key <= MAX_KEY_CODE) then
110   begin
111     tmpInterval := MilliSecondsBetween(Time(),KeyDown[Key]);
112     KeyPress[Key,KeyIndex[Key]] := tmpInterval;
113     inc(KeyIndex[Key]);
114     if KeyIndex[Key] > MAX_SER_LEN then
115       KeyIndex[Key] := 1;
116   end;
117 end;
```

Рисунок 4.6 — Облік таймінгу для зняття сигнатур відпускання клавіші

У фрагменті коду на рисунку 4.7 зображено обчислення статистики, де розраховується середньоарифметичне:

1. час до натиснення певної клавіші;
2. час до утримання певної клавіші;
3. час до натискання клавіші;

4. час утримання клавіші.

Після обчислення дані зберігаються в базі в таблиці «timing» та «user» та можуть бути завантажені для поточного користувача.

```
121 procedure CalcStatistic;
122 var
123   tmpKey:integer;
124   i: integer;
125   tmpKeyCount: integer;
126 begin
127   tmpKeyCount := 0;
128   KeyPressTOvr := 0;
129   for tmpKey := 1 to MAX_KEY_CODE do
130   begin
131     KeyPressOvr[tmpKey] := 0;
132     KeyIntOvr[tmpKey] := 0;
133     if KeyCount[tmpKey] > 0 then
134     begin
135       i := 1;
136       while (i <= KeyCount[tmpKey]) and (i<=MAX_SER_LEN) do
137       begin
138         KeyPressOvr[tmpKey] := KeyPressOvr[tmpKey] + KeyPress[tmpKey,i];
139         KeyIntOvr[tmpKey] := KeyIntOvr[tmpKey] + KeyInt[tmpKey,i];
140         inc(i);
141       end;
142       dec(i);
143       KeyPressOvr[tmpKey] := KeyPressOvr[tmpKey] div i;
144       KeyIntOvr[tmpKey] := KeyIntOvr[tmpKey] div i;
145       KeyPressTOvr := KeyPressTOvr + KeyPressOvr[tmpKey];
146       KeyIntTOvr := KeyIntTOvr + KeyIntOvr[tmpKey];
147       inc( tmpKeyCount );
148     end;
149   end;
150   if tmpKeyCount<>0 then
151   begin
152     KeyPressTOvr := KeyPressTOvr div tmpKeyCount;
153     KeyIntTOvr := KeyIntTOvr div tmpKeyCount;
154   end;
155 end;
```

Рисунок 4.7 — Обчислення статистики

На рисунку 4.8 зображено ключову функцію програми – порівняння сигнатур, де окремо порівнюються сигнатури для пароля та сигнатури при наборі

тексту, так як склад символів в цих випадках відрізняється, що може відбиватися на динаміці набору тексту.

```
function CheckSign(AType:integer):boolean;
var
  tmpDiv: double;
  tmpDiv2: double;
  tmpDivMax: double;
  tmpCount: integer;
  tmpKey:integer;
begin
  CalcStatistic;
  tmpDivMax := 0;
  AddToLog('Перевірка сигнатур. Тип перевірки: '+inttostr(AType));
  tmpDiv := abs(UserKeyPressTOvr - KeyPressTOvr) / UserKeyPressTOvr;
  if tmpDiv > tmpDivMax then
    tmpDivMax := tmpDiv;
  AddToLog('Відхилення по утриманню: '+FloatToStr(tmpDiv));
  tmpDiv := abs(UserKeyIntTOvr - KeyIntTOvr) / UserKeyIntTOvr;
  if tmpDiv > tmpDivMax then
    tmpDivMax := tmpDiv;
  AddToLog('Відхилення по темпу: '+FloatToStr(tmpDiv));
  tmpDiv := 0;
  tmpDiv2 := 0;
  tmpCount := 0;
  for tmpKey := 1 to MAX_KEY_CODE do
  begin
    if (KeyPressOvr[tmpKey]<>0) and (KeyIntOvr[tmpKey]<>0) and (UserKeyPressOvr[tmpKey]<>0) and (UserKeyIntOvr[tmpKey]<>0) then
    begin
      tmpDiv := tmpDiv + abs(UserKeyIntOvr[tmpKey] - KeyIntOvr[tmpKey]) / UserKeyIntOvr[tmpKey];
      tmpDiv2 := tmpDiv2 + abs(UserKeyPressOvr[tmpKey] - KeyPressOvr[tmpKey]) / UserKeyPressOvr[tmpKey];
      inc(tmpCount);
    end
  end;
  if tmpCount>0 then
  begin
    tmpDiv := tmpDiv / tmpCount;
    tmpDiv2 := tmpDiv2 / tmpCount;
  end;
  AddToLog('Середнє відхилення по утриманню по клавішах: '+FloatToStr(tmpDiv));
  AddToLog('Середнє відхилення по темпу по клавішах: '+FloatToStr(tmpDiv2));
  if tmpDiv > tmpDivMax then
    tmpDivMax := tmpDiv;
  if tmpDiv2 > tmpDivMax then
    tmpDivMax := tmpDiv2;
  result := tmpDivMax < SQLExecute('select coalesce(k_dev,0) from param');
  if not result then
    AddToLog('Виявлено невідповідність психомоторних характеристик поточного користувача і реєстраційних даних');
end;
```

Рисунок 4.8 — Порівняння сигнатур

При порівнянні розраховуються середні значення по поточним відомостям про дії користувача. В процесі порівняння знаходиться максимальне відхилення характеристик. Порівнюються максимальне відхилення з максимально допустимим відхиленням, яке зберігається в налаштуваннях програми.

Генерація випадкового пароля показана на рисунку 4.9, де вказано, що довжина пароля буде не менше мінімальної, але може відрізнятися на випадкову

величину від 0 до VAR_PASS_LENGTH. Для побудови пароля береться набір допустимих символів, який зберігається в налаштуваннях програми та в таблиці «param».

```
287 procedure frmChangePass_brnCreatePass_OnClick (Sender: TObject; var Cancel: boolean);
288 var
289   tmpPassLength: integer;
290   tmpPass: string;
291   tmpValidKeys: string;
292   i: integer;
293   j: integer;
294 begin
295   tmpPassLength := MIN_PASS_LENGTH + trunc(Random()*(VAR_PASS_LENGTH+1));
296   tmpValidKeys := SQLExecute('select coalesce(ValidPasskey,''!'') from param');
297   tmpPass := '';
298   for i:=1 to tmpPassLength do
299   begin
300     tmpPass := tmpPass + tmpValidKeys[ trunc(Random()*length(tmpValidKeys))+1 ];
301   end;
302   frmChangePass.edtPassword.Text := tmpPass;
303   frmChangePass.edtEnterPass.Text := '';
304   frmChangePass.btnConfirm.Enabled := true;
305   InitTimings;
306 end;
```

Рисунок 4.9 — Генерація випадкового пароля

```
313 procedure frmDataEdit_Button1_OnAfterClick (Sender: TObject);
314 begin
315   if not CheckSign(TT_TEXT) then
316   begin
317     ShowMessage('Виявлено розбіжність психомоторних характеристик поточного користувача і реєстраційних даних. Доступ до програми заблокований. Будь ласка, пройдіть повторну авторизацію!');
318     frmMain.Close;
319   end;
320 end;
```

Рисунок 4.10 — Контроль збереження даних

4.9 Реалізація системи ідентифікації користувачів на основі клавіатурного почерку

Програма складається файлів та директорій зображених на рисунку 4.11.

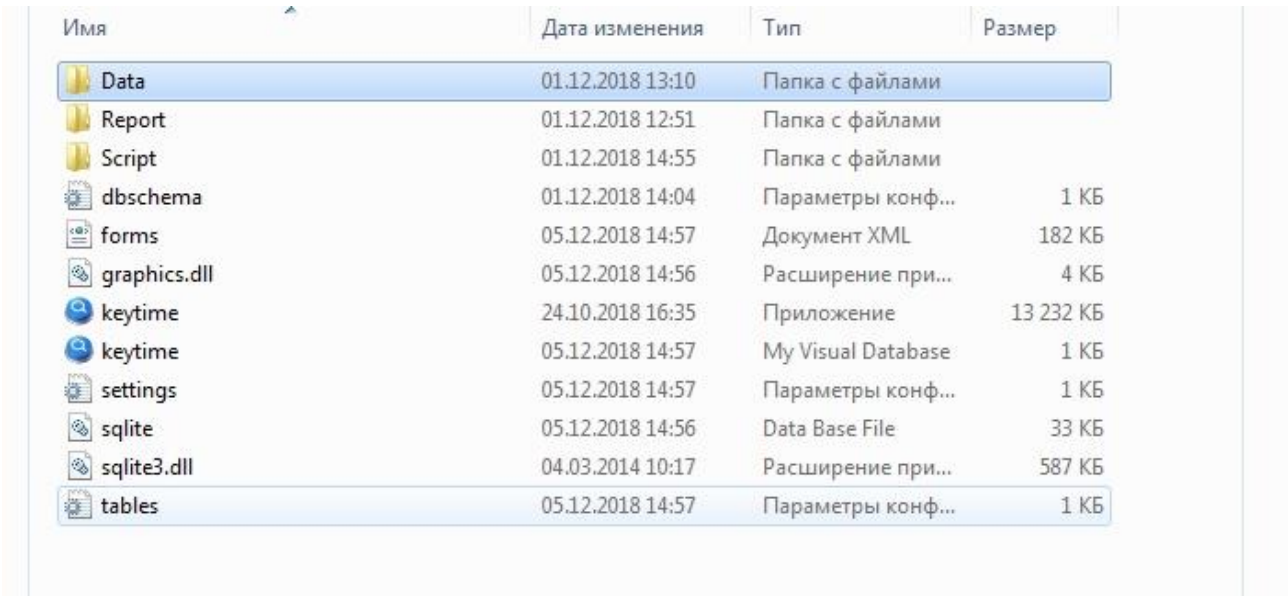


Рисунок 4.11 — Склад файлів програми

1. Data – папка з додатковими графічними ресурсами, що використовуються в даній програмі.
2. dbschema.ini – файл створений для налаштування відображення діаграми даних.
3. forms.xml – файл з налаштуванням властивостей форм і інших компонентів графічного інтерфейсу.
4. graphics.dll – бібліотека з додатковими графічними ресурсами, що були використані в програмі.
5. keytime.exe – виконуваний модуль, що був створений на платформі My Visual Database.
6. keytime.vdb – данні про проект.
7. Report – папка, яка була створена для звітів.
8. Script – папка для вихідних та скомпільованих скриптів.
9. settings.ini – файл для налаштування програми.
10. sqlite.db – база даних.
11. sqlite3.dll – бібліотека для роботи з базою даних.
12. tables.ini – файл структури бази даних.

Під час першого запуску програми в базі створюється обліковий запис для адміністратора, який має логін admin, та пароль admin. Набираємо видані данні, та нажимаємо «Вхід», як зображено на рисунку 4.12.

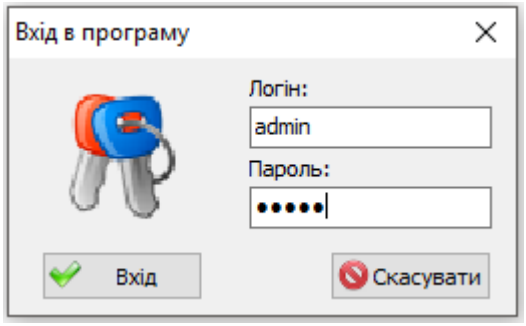


Рисунок 4.12 — Вигляд вікна входу із заповненими даними для адміністратора

Нижче на рисунку 4.13 показано основну форму для адміністратора. В ній відображається журнал дій та кнопки для налаштування програми:

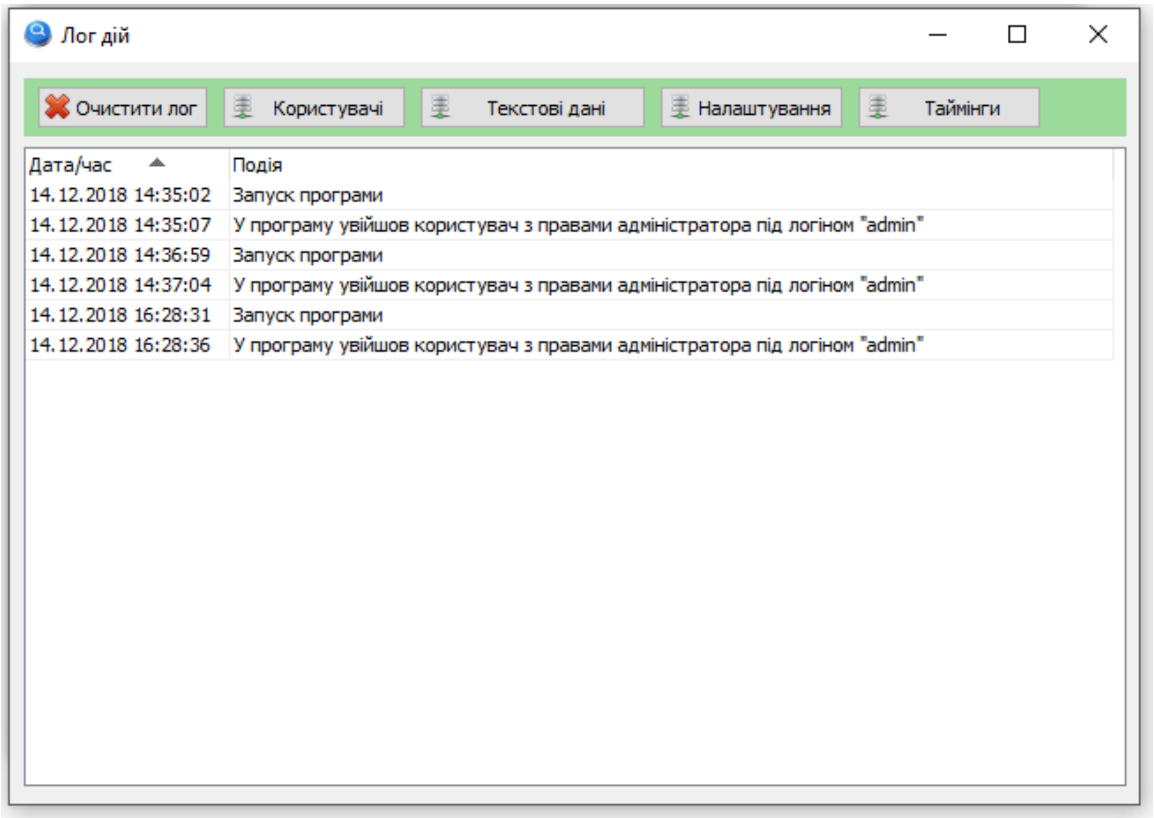


Рисунок 4.13 — Основна форма для адміністратора

1. Очистити журнал – видалити всі дані подій та часу з вікна логу.
2. Користувачі – ведення облікових записів створених користувачів.
3. Текстові дані – створення текстових даних для отримання сигнатур ідентифікації користувачів при їх реєстрації.
4. Налаштування – загальні настройки програми.
5. Таймінги – відображення сигнатур користувачів, що успішно пройшли всі етапи реєстрації.

На рисунку 4.14 приведено журнал реєстрації користувачів та форма редагування.

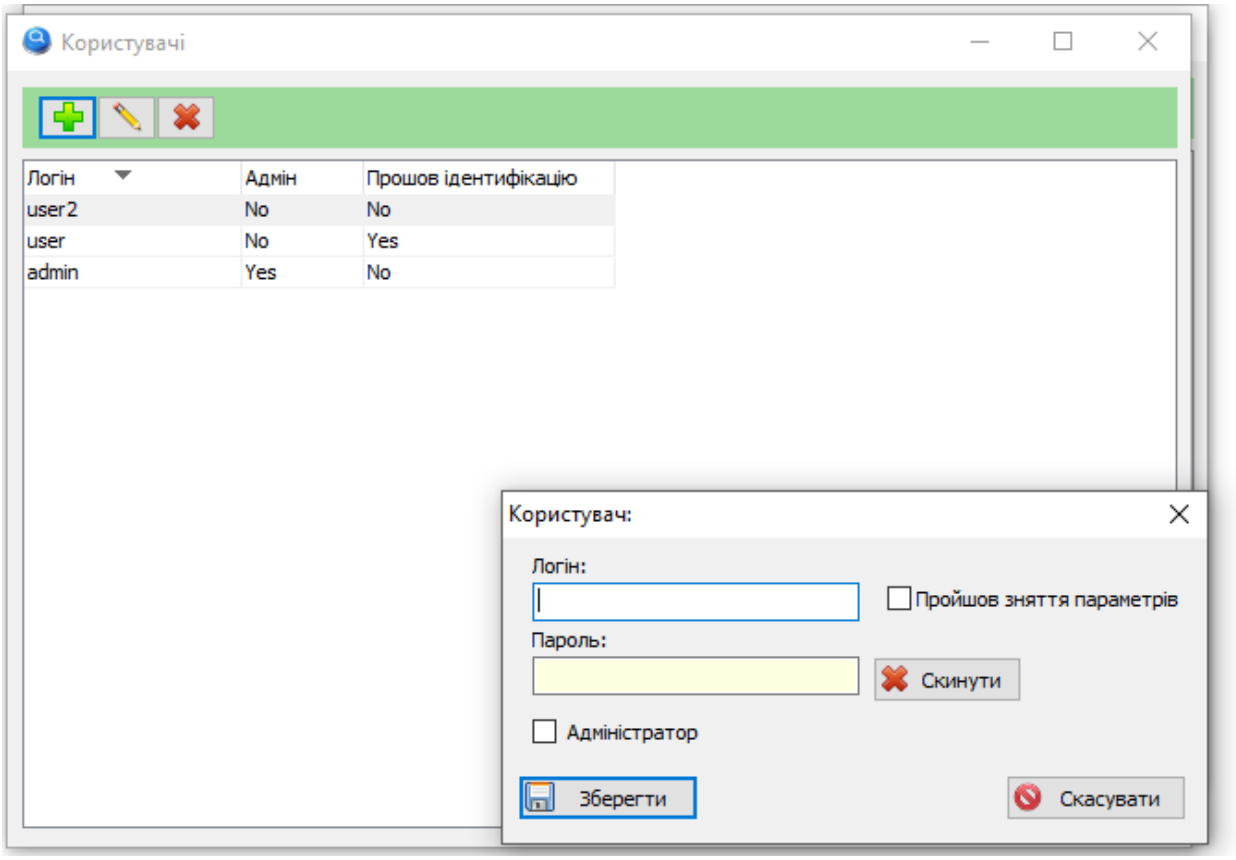


Рисунок 4.14 — Журнал користувачів

При створенні нового користувача, адміністратором задається логін. Якщо відмітити чекер «Адміністратор», то додаткової ідентифікації по сигнатурам проводитись не буде. Чекер «Пройшов зняття параметрів» встановлюється

автоматично після того, як обраний користувач ввів текстову частину ідентифікації. Кнопка «Скинути» очищує поле пароля для користувача. Пароль вводить користувач, але у випадку його втрати – адміністратор може його скинути для генерації користувачем нового пароля. Якщо у користувача пароль відсутній, то при першому вході в програму він буде змушений самостійно згенерувати пароль за допомогою програми та пройти ідентифікацію.

Вікно текстових даних, що зображено на рисунку 4.15 – це набір текстів, які випадковим чином надаються новому користувачеві для введення під час проходження реєстрації. Порядкові номера текстів повинні бути послідовними, від 1 до N, без пропусків, тому що алгоритм обирає їх за допомогою генератора випадкових чисел.

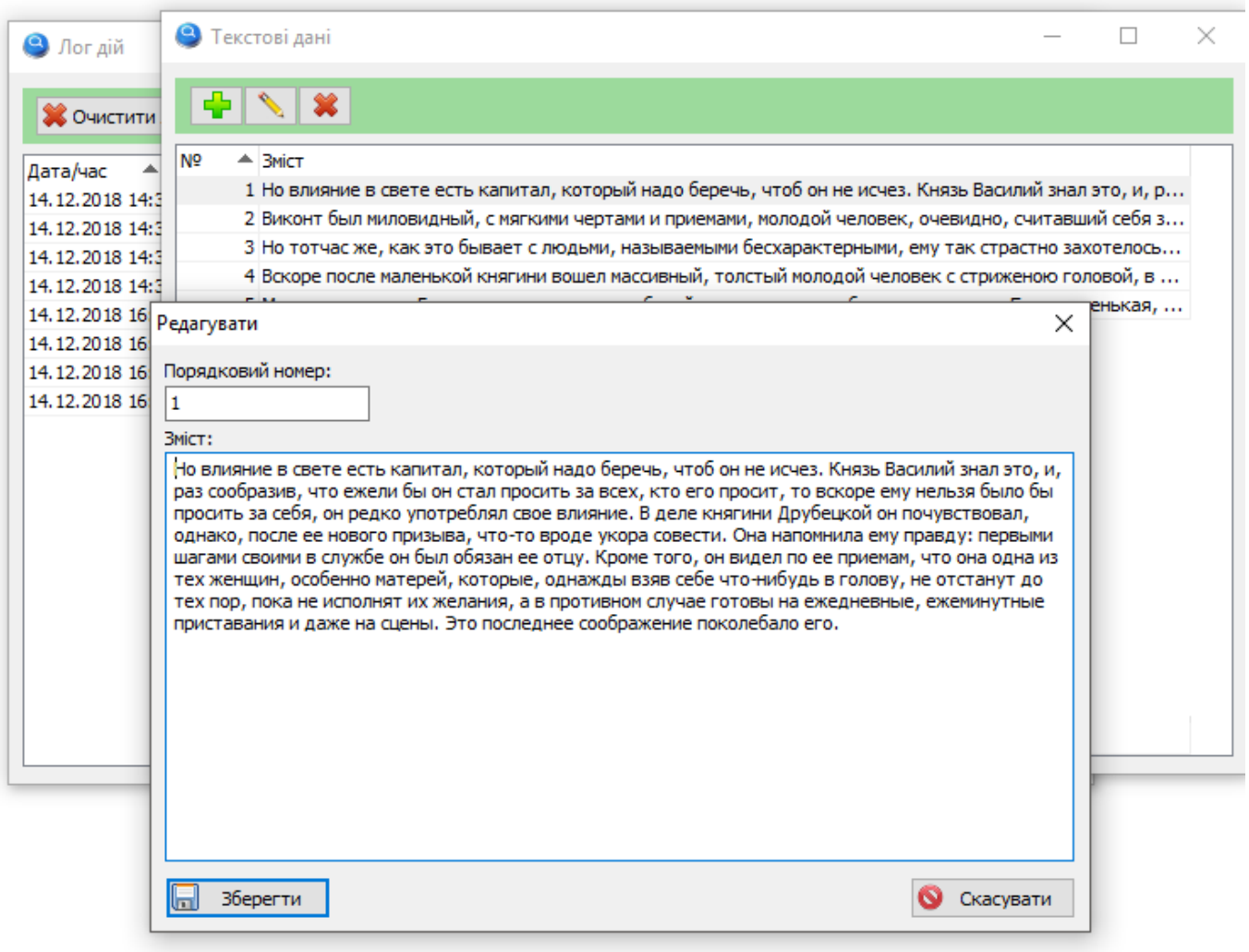


Рисунок 4.15 — Вікно текстових даних для реєстрації

Форма з параметрами, яка зображена на рисунку 4.16, створена для налаштування параметрів та має наступні поля:

1. Поріг спрацювання – коефіцієнт відхилення, який є порогом для алгоритму порівняння поточних та отриманих при реєстрації сигнатур. При перевищенні даного порогу програма блокує користувача та вимагає повторної авторизації.
2. Кількість повторень при введені пароля – експериментально підібране число для кількості повторень паролю при реєстрації. Якщо це число маленьке, то похибка вимірювань велика, але якщо число дуже велике, то похибка також збільшується, адже тимчасово спрацьовує м'язова пам'ять, через яку показники змінюються, але з часом користувач втрачає цю здатність.
3. Доступні символи для пароля – пароль генерується тільки з символів, які представлені в цьому полі.

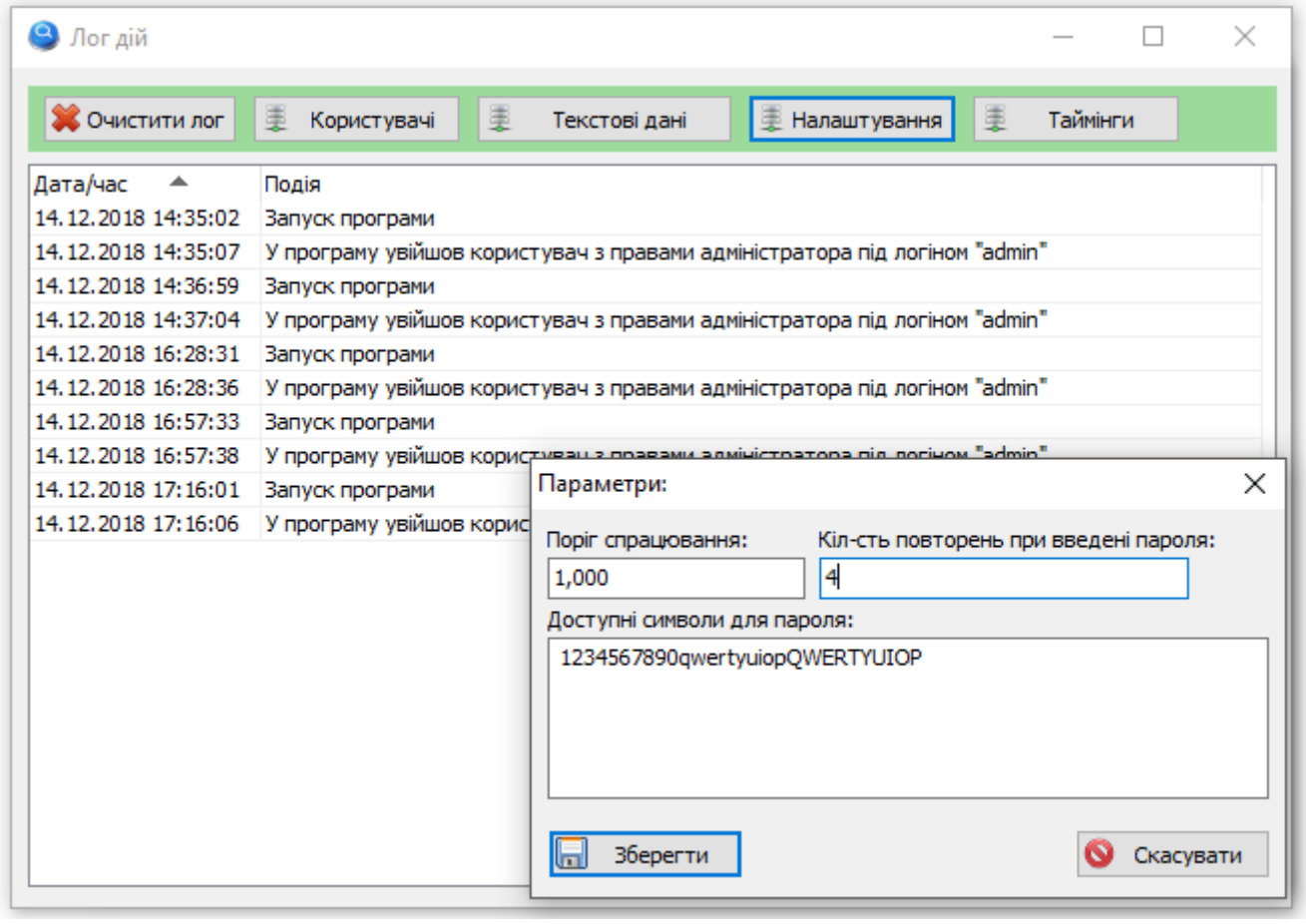
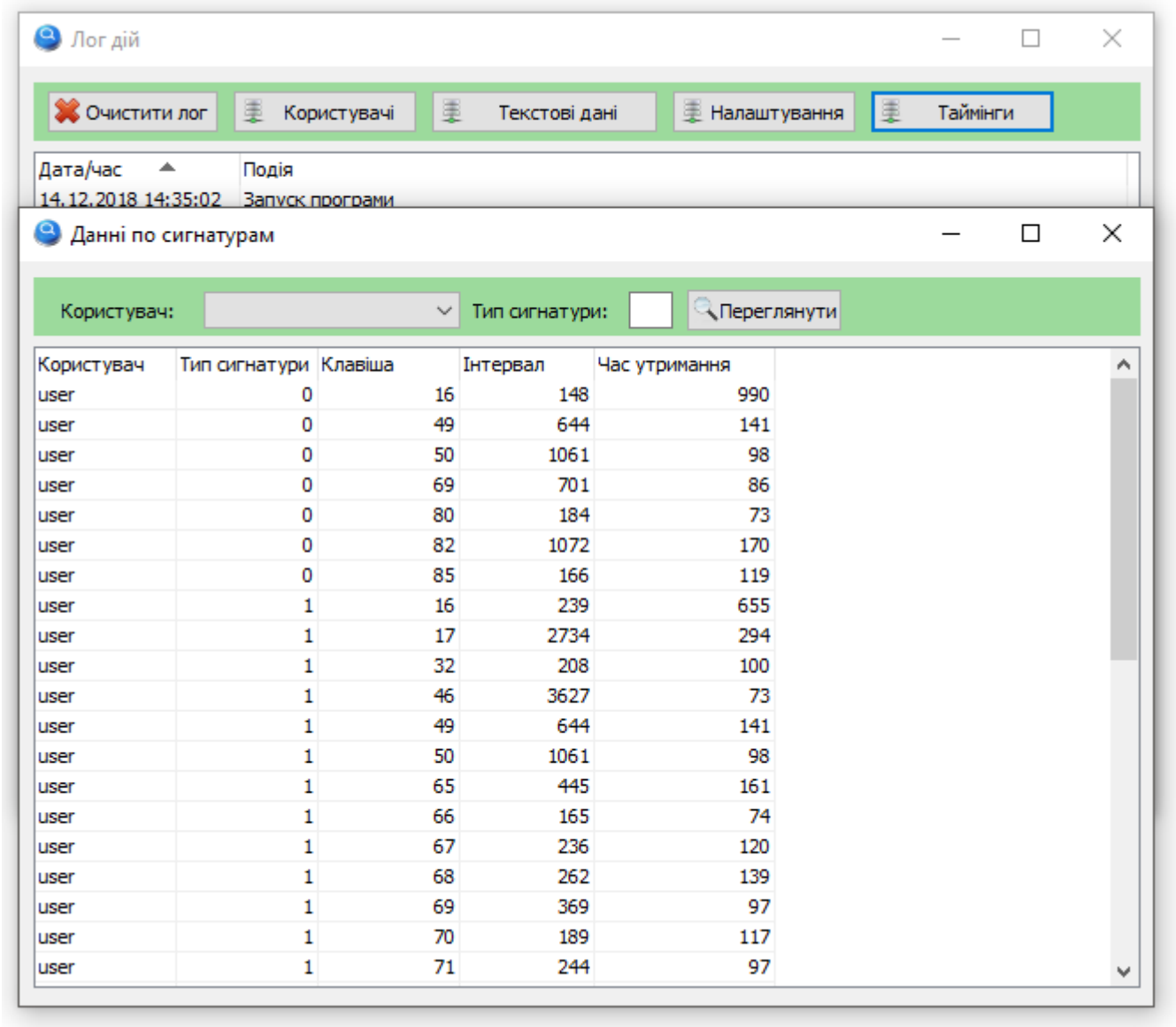


Рисунок 4.16 — Вікно форми за параметрами

Вікно таймінгів – це середні значення, що були отримані при реєстрації користувача під час введення пароля та контрольного тексту (рисунок 4.17):



Користувач	Тип сигнатури	Клавіша	Інтервал	Час утримання
user	0	16	148	990
user	0	49	644	141
user	0	50	1061	98
user	0	69	701	86
user	0	80	184	73
user	0	82	1072	170
user	0	85	166	119
user	1	16	239	655
user	1	17	2734	294
user	1	32	208	100
user	1	46	3627	73
user	1	49	644	141
user	1	50	1061	98
user	1	65	445	161
user	1	66	165	74
user	1	67	236	120
user	1	68	262	139
user	1	69	369	97
user	1	70	189	117
user	1	71	244	97

Рисунок 4.17 — Вікно таймінгів користувачів, що зареєструвалися

Таким чином в повноваження Адміністратора входить:

1. Ведення журналу користувачів.
2. Перегляд журналу подій.
3. Налаштування параметрів програми.

На відміну від адміністратора, у звичайного користувача під час першого входу пароля немає, а є тільки логін, що був виданий адміністратором при створенні нового облікового запису користувача (рисунок 4.18).

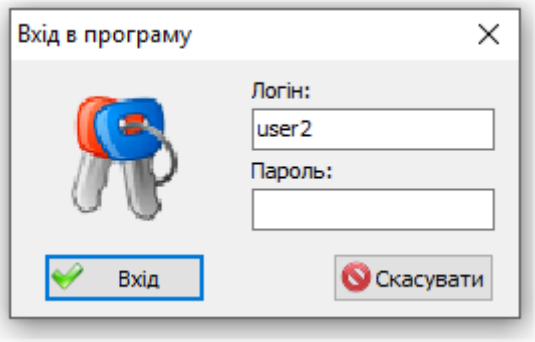


Рисунок 4.18 — Вигляд вікна першого входу звичайного користувача

Після натиснення кнопки «Вхід» отримуємо повідомлення про те, що обліковий запис потребує встановлення пароля та переходимо до вікна генерації нового пароля, яке зображене на рисунку 4.19.

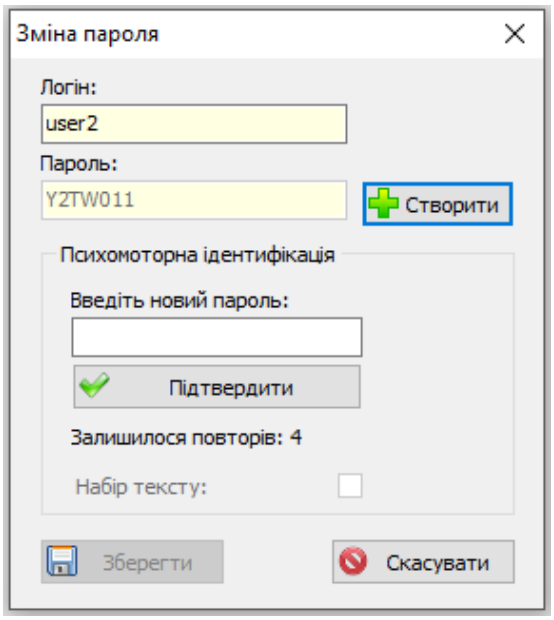


Рисунок 4.19 — Вікно для генерації нового паролю

Натискаємо кнопку «+ Створити» для того, щоб програма згенерувала пароль з набору допустимих символів, які були вказані в налаштуваннях програми адміністратором. Тобто, користувач не набирає його сам, але може обрати собі той, що більше сподобався, натиснувши кнопку «+Створити» декілька разів. Потім, користувач повинен ввести пароль указану кількість разів і натиснути кнопку «Підтвердити». В процесі введення пароля фіксуються таймінги динаміки набору.

Після закінчення введення пароля з'являється вікно за повідомленням, що користувачеві потрібно пройти додаткову ідентифікацію (рисунок 4.20).

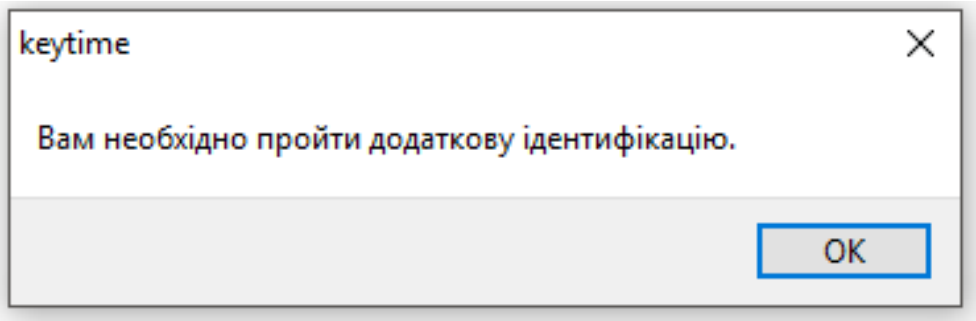


Рисунок 4.20 — Вікно з повідомленням про потребу додаткової ідентифікації при реєстрації нового користувача

Після натиснення кнопки «Ок», з'являється вікно, в якому відображений випадково згенерований текст з тих варіантів, які були створені адміністратором в текстових даних програми (рисунок 4.21). Важливо те, що користувач не має можливості скопіювати текст, а повинен вводити його руками, щоб програма могла правильно отримати таймінги. Якщо текст був введений без помилок, то таймінги записуються в базу та в вікні створення паролю ставиться відмітка про те, що текст був набраний та користувач пройшов додаткову ідентифікацію.

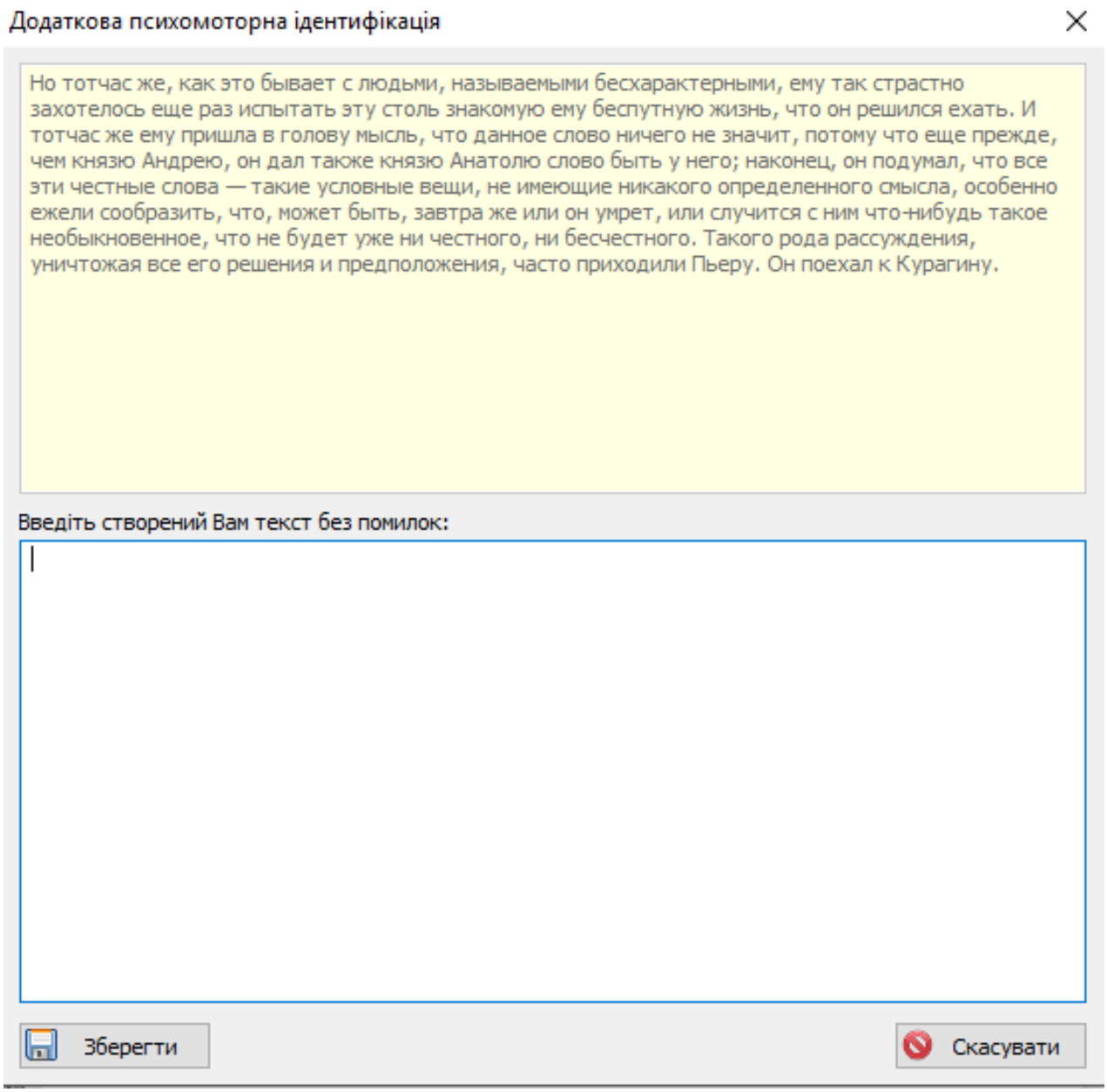


Рисунок 4.21 — Вікно з додатковою ідентифікацією для користувача по вільному тексту

Якщо під час входу користувач правильно ввів пароль, та його клавіатурний почерк співпадає з тим, що був отриманий під час реєстрації, то він попадає в основне вікно програми, де він може набирати текст.

Під час написання тексту в щоденник користувачем (рисунок 4.22), також ведеться приховане спостереження за динамікою набору символів, та в момент збереження даних здійснюється порівняння з еталоном, який закріплений за даним

користувачем в базі. В випадку розходжень з’являється повідомлення про те, що в доступі відмовлено і програма закривається.

Таким чином ведеться як первинне розпізнавання, так і спостереження в ході роботи програми.

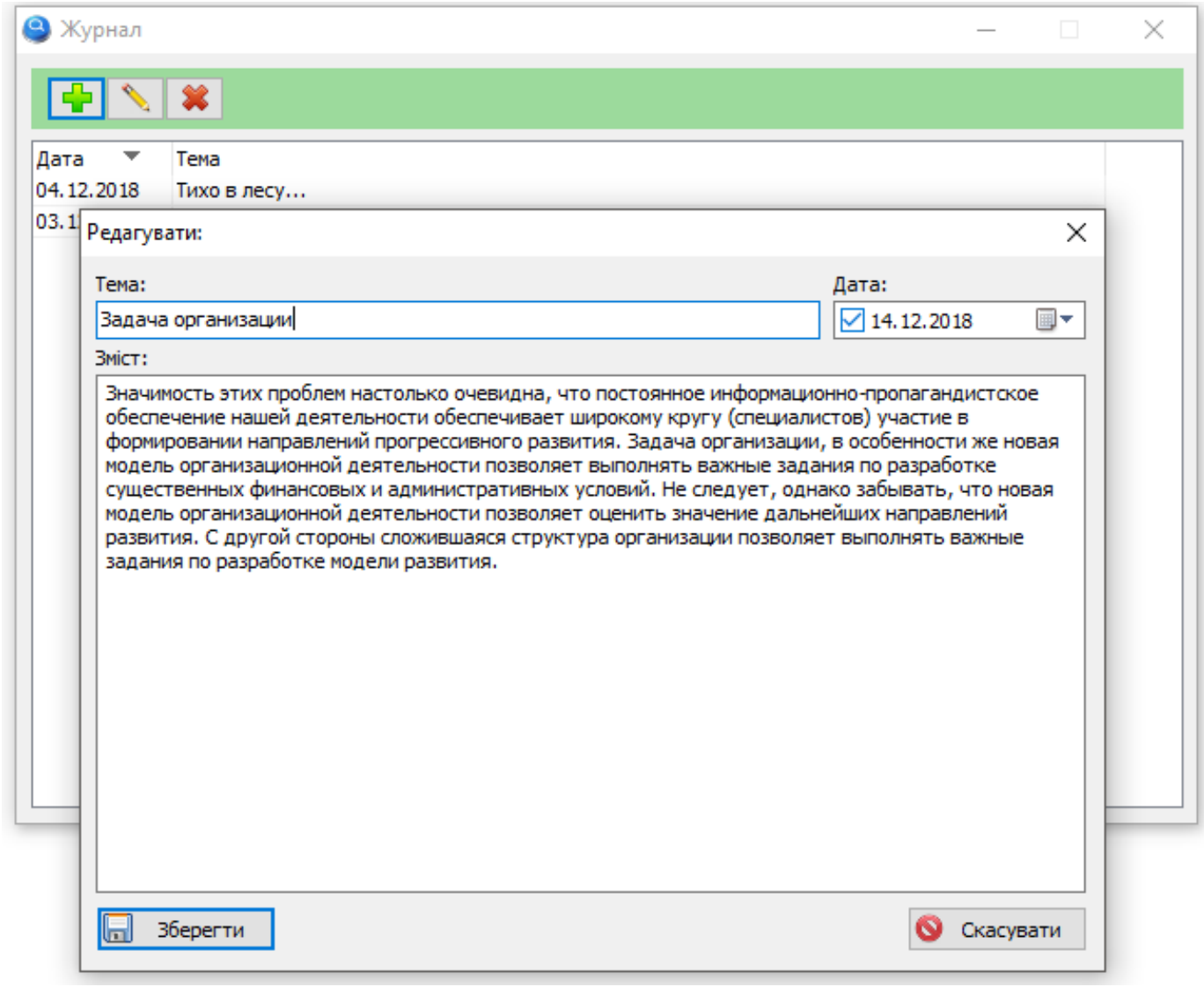


Рисунок 4.22 — Щоденник користувача

5 РОЗРОБЛЕННЯ СТАРТАП-ПРОЕКТУ

5.1 Опис ідеї проекту

Основною метою роботи є розробка програми, що відноситься до засобів ідентифікації користувачів. Дана технологія базується на основі клавіатурного почерку та здатна забезпечити підвищену безпеку даних та точність встановлення легітимного користувача, у порівнянні з іншими способами перевірки відповідності, в інформаційних системах.

Таблиця 5.2 — Опис ідеї стартап-проекту

Зміст ідеї	Напрямки застосування	Вигоди для користувача
Ідентифікація користувачів на основі клавіатурного почерку.	1.Ідентифікація користувача в системі.	Відносна дешевизна, порівняно з існуючими закордонними аналогами.
	2.Прихований моніторинг.	Можливість спостереження за роботою співробітників.
	3.Наукова діяльність.	Застосування новітніх технологій.

При використанні звичайної клавіатури, простої технології та застосування алгоритмів автоматизації робить програмне забезпечення конкурентоспроможним.

Відомо, що в Україні існують тільки закордонні аналоги, отже наявність вітчизняного продукту, що є дешевшим, не гіршим за аналоги та допомагає в роботі, є дуже перспективним.

Таблиця 5.3 — Визначення сильних, слабких та нейтральних характеристик ідеї проекту.

№ п/п	Техніко-економічні характеристики ідеї	(потенційні) товари/концепції конкурентів				W (слабка сторона)	N (нейтральна сторона)	S (сильна сторона)
		Мій проект	Daon	Experian	Fujitsu			
1.	Вартість ПЗ	Низька	Висока	Висока	Висока			+
2.	Час обробки	Низький	Високий	Низький	Низький		+	
3.	Автоматизація	70 %	80%	70%	60%		+	
5.	Споживачі (відомий бренд)	-	-	+	+	+		

5.2 Технологічний аудит ідеї проекту

Таблиця 5.4 — Технологічна здійсненність ідеї проекту

№ п/п	Ідея проекту	Технології її реалізації	Наявність технологій	Доступність технологій
1.	Ідентифікація користувачів	MyVisualDatabase дає змогу використовувати даний продукт в	Наявні.	Так.

		розповсюдженій системі Windows.		
2.	Алгоритм прихованого моніторингу	Критерії порівняння параметрів клавіатурного почерку.	Наявні.	Так.
Обраною технологією є ідентифікація користувачів та алгоритм прихованого моніторингу із реалізацією у вигляді програми. Для належної роботи програми необхідно зареєструватися користувачеві для запису параметрів клавіатурного почерку в шаблон.				

5.3 Аналіз ринкових можливостей запуску стартап-проекту

Головною перевагою та характеристикою даного продукту є співвідношення ціни та можливостей, що відповідають високому рівню якості. Зависока ціна іноземних аналогів та їх недоступність для широкого кола споживачів робить даний товар перспективним для застосування як в Україні, так і закордоном.

Таблиця 5.5 — Попередня характеристика потенційного ринку стартап-проекту

№ п/п	Показники стану ринку (найменування)	Характеристика
1	Кількість головних гравців, од.	0. Тільки закордонні, немає аналогів в Україні.
2	Загальний обсяг продаж, ум.од/час	Поодиничний продаж, закордонних товарів. В середньому – 400 доларів

		за річний доступ.
3	Динаміка ринку (якісна оцінка)	Зростає
4	Наявність обмежень для входу (вказати характер обмежень).	Новітня технологія потребує ресурсів, практичної перевірки та доробки універсального алгоритму.
5	Специфічні вимоги до стандартизації та сертифікації.	Не потребує сертифікації.
6	Середня норма рентабельності в галузі (по ринку), %	50

Основною споживчою аудиторією є установи, які мають важливу, секретну інформацію та для роботи користувачів потрібний постійний прихований моніторинг для ідентифікації користувачів. Розповсюдження програмного забезпечення відбувається через дистриб'юторів або представників. Алгоритм є універсальним, без залежності від рекомендацій установи та сфери діяльності.

Таблиця 5.5 — Характеристика потенційних клієнтів стартап-проекту

№ п/п	Потреба, що формує ринок	Цільова аудиторія (цільові сегменти ринку)	Відмінності у поведінці різних потенційних цільових груп клієнтів
1. Дешевизна алгоритму	Зручний інструмент для підвищення безпеки інформаційних систем.	Зацікавлені в купівлі дешевшого.	Задовільна цінова політика, відповідність ДСТУ/ ISO, сумісність з ПК, висока якість, гарні технічні характеристики (час
2.Зменшення ризику		Зацікавлені в кращій якості.	

виникнення людського фактору.			обробки, точність).
3. Купувати вітчизняне		Зацікавлені в підтримці національного виробника.	
4.Отримати закордонний аналог по якості за українською ціною		Зацікавлені за менші гроші отримати високоякісний матеріал.	

Для досягнення необхідного рівня конкурентоспроможності, що являється основною загрозою для інших виробників такої продукції, потрібна гідна реклама. Також, так як технологія є новою та унікальною, важливим пунктом є ретельний підбір членів команди.

Таблиця 5.6 — Фактори загроз

№ п/п	Фактор	Зміст загрози	Можлива реакція компанії
1.	Конкурентоспроможність	Новий товар на ринку– невідомий бренд–низька конкурентоспроможність	Реклама, вдалий маркетинговий проект, залучення дилерів, спонсорів

			до співпраці.
2.	Невідомий бренд	Невідомий новий товар невідомої фірми.	Розкрутка товару.
3.	Комерційна таємниця	Можливість відкриття технології невідомими робітниками.	Підписання строгих контрактів з несенням матеріальної компенсації.

Найголовнішою можливістю є залучення іноземних спонсорів та можливостей реалізації товару закордоном.

Таблиця 5.7 — Фактори можливостей

№ п/п	Фактор	Зміст можливості	Можлива реакція компанії
1.	Залучення іноземних спонсорів.	Залучення іноземних капіталів, підтримка спонсорів.	Укладення договорів про співпрацю та налагодження торгових контактів.
2.	Налагодження контактів з державними установами та приватними підприємцями.	Залучення дистриб’юторів.	Пошук та налагодження контактів з компаніями, державними установами та приватними підприємцями.

Таблиця 5.8 — Ступеневий аналіз конкуренції на ринку

Особливості конкурентного середовища	В чому проявляється дана характеристика	Вплив на діяльність підприємства (можливі дії компанії, щоб бути конкурентоспроможною)
1. Вказати тип конкуренції - монополія/олігополія/ монополістична/чиста	Монополія, на даний час в Україні немає аналогів.	Розкрутка національного товару, залучення інвесторів.
2. За рівнем конкурентної боротьби	Міжнаціональний	Випуск аналогу закордонних продуктів, дешевша ціна на аналогічну продукцію.
3. За галузевою ознакою	Внутрішньогалузева	Використовується в багатьох сферах, де потребується ідентифікація користувачів.
4. Конкуренція за видами товарів	Товарно-родова	Власні розробки, унікальна технологія, реклама.
5. За характером конкурентних переваг	Цінова	Ціна нижча за іноземні аналоги.
6. За інтенсивністю - марочна/не марочна	Не марочна	Реклама, покращення технологій

Таблиця 5.9 — Аналіз конкуренції в галузі за М. Портером

	Прямі конкуренти в галузі	Потенційні конкуренти	Постачальники	Клієнти	Товари-замінники
Складові аналізу	Немає, або закордонні	Гнучкі ціни, розмір капіталовкладень	Велика концентрація постачальників	Якісна продукція, не потребує великої кількості товару	Вища ціна, вищі змінні витрати
Висновки	Невисока конкурентна боротьба	Є можливості входу в ринок, наявні потенційні конкуренти	Постачальники диктують умови роботи на ринку, наприклад, ціну та швидкість розповсюдження	Так, залежно від попиту на товар.	Обмежень немає

Робота на ринку є можливою, але дуже залежить від матеріальної спроможності установ, потенційних клієнтів. Так як вони купують набагато дорожчий товар закордонних виробників, можна зробити висновок про те, що даний товар буде користуватись періодичним попитом. Також, важливим є фактор розповсюдження шляхом постачальників. Конкурентна боротьба є високою, проте лише з закордонними аналогами, адже таких в Україні немає.

Таблиця 5.10 — Обґрунтування факторів конкурентоспроможності

№ п/п	Фактор конкурентоспроможності	Обґрунтування (наведення чинників, що роблять фактор для порівняння конкурентних проектів значущим)
1	Час	Менша необхідність машинної праці та економія часу.
2	Ціна	Ґрунтується на собівартості-отже є нижчою.
3	Якість	Не гірша за закордонні аналоги.

Таблиця 5.11 — Порівняльний аналіз сильних та слабких сторін власного проекту

№ п/п	Фактор конкурентоспроможності	Бали 1-20	Рейтинг товарів-конкурентів у порівнянні з власною компанією						
			–3	–2	–1	0	+1	+2	+3
1	Час	15				+			
2	Ціна	18	+						
3	Якість	14						+	

Таблиця 5.12 — SWOT-аналіз стартап-проекту

Сильні сторони: Собівартість Ціна Рентабельність	Слабкі сторони: Невідомий бренд Великі капіталовкладення Необхідність встановленого додаткового ПЗ
Можливості: Вихід на закордонний ринок Забезпечення споживчих потреб Дохід	Загрози: Викриття комерційної таємниці Недостатня реалізація

Таблиця 5.13 — Альтернативи ринкового впровадження стартап-проекту

№ п/п	Альтернатива (орієнтовний комплекс заходів) ринкової поведінки	Ймовірність отримання ресурсів	Строки реалізації
1	Практичне використання алгоритму, вдосконалення	+	1 рік
2	Залучення іноземних фахівців	-	1 рік
3	Рекламна кампанія	+	1,5 року
4	Отримання міжнародних сертифікатів	+	5 років
5	Вихід на закордонний ринок	+	6 років

Обраною альтернативою є залучення рекламної кампанії, розкрутка нової технології.

5.4 Розроблення ринкової стратегії проекту

Таблиця 5.14 — Вибір цільових груп потенційних споживачів

№ п/п	Опис профілю цільової групи потенційних клієнтів	Готовність споживачів сприйняти продукт	Орієнтовний попит в межах цільової групи (сегменту)	Інтенсивність конкуренції в сегменті	Простота входу у сегмент
1	Банки	Не готові	Дуже необхідно	Мінімальна	Важка
2	Державні установи, бюджетні організації	Готові	Необхідно	Мінімальна	Легка
Які цільові групи обрано: державні установи, бюджетні організації.					

Таблиця 5.15 — Визначення базової стратегії розвитку

№ п/п	Обрана альтернатива розвитку проекту	Стратегія охоплення ринку	Ключові конкурентоспроможні позиції відповідно до обраної альтернативи	Базова стратегія розвитку
1	Банки	Стратегія лідерства по витратах	– ціна – простота застосування – продуктивність роботи	Ретельний контроль за постійними витратами, зниження виробничих,

			– низькі витрати	збутових і рекламних витрат, проведення інвестиції, спрямованої на зменшення витрат, ретельне опрацювання алгоритму нових товарів.

Таблиця 5.16 — Визначення базової стратегії конкурентної поведінки

№ п/п	Чи є проект «першопрохідцем» на ринку?	Чи буде компанія шукати нових споживачів, або забирати існуючих у конкурентів?	Чи буде компанія копіювати основні характеристики товару конкурента, і які?	Стратегія конкурентної поведінки
1	Так	Буде шукати	Ні, продукт застосовує власний алгоритм для ПЗ.	Стратегія наслідування лідера

Таблиця 5.17 — Визначення стратегії позиціонування

№ п/п	Вимоги до товару цільової аудиторії	Базова стратегія розвитку	Ключові конкурентоспроможні позиції власного стартап-проекту	Вибір асоціацій, які мають сформувати комплексну позицію власного проекту (три ключових)
1	Дешевизна закупівлі підтримки ПЗ.	Стратегія наслідування лідера	– ціна – простота виготовлення – кількість виробленої продукції – низькі витрати на сировину	– зниження цін – вітчизняне виробництво – невеликі обсяги – універсальність для кожного пацієнта
2	Зменшення ризику людської помилки.			
3	Купувати вітчизняне.			
4	Отримати закордонний аналог по якості за українською ціною.			

5.5 Розроблення маркетингової програми стартап-проекту

Таблиця 5.18 — Визначення ключових переваг концепції потенційного товару

<table>
<tr><td>Рівні товару</td><td colspan="3">Сутність та складові</td></tr>
<tr><td>1.Товар за задумом</td><td colspan="3">Програмне забезпечення надає змогу швидко ідентифікувати користувача та виявити заміну користувача.</td></tr>
<tr><td rowspan="4">2.Товар у реальному виконанні</td><td>Властивості/характеристики</td><td>М/Нм</td><td>Вр/Тх /Тл/Е/Ор</td></tr>
<tr><td>1. Висока швидкість
2.Точність
3. Сумісний з більшістю ПК</td><td></td><td></td></tr>
<tr><td colspan="3">Якість: стандарти, нормативи, параметри тестування тощо
Стандартизація відповідно до ДСТУ,ISO.
Регламентується НД, СРМ.</td></tr>
<tr><td colspan="3">Пакування відсутнє.</td></tr>
<tr><td colspan="4">Марка: назва організації-розробника + назва товару</td></tr>
<tr><td colspan="4">Потенційний товар буде захищено від копіювання: патентування, сертифікати відповідності.</td></tr>
</table>

Таблиця 5.19 — Опис трьох рівнів моделі товару

<table>
<tr><td>Рівні товару</td><td colspan="3">Сутність та складові</td></tr>
<tr><td>1.Товар за задумом</td><td colspan="3">Програмне забезпечення надає змогу швидко ідентифікувати користувача та виявити заміну користувача.</td></tr>
<tr><td rowspan="3">2.Товар у реальному виконанні</td><td>Властивості/характеристики</td><td>М/Нм</td><td>Вр/Тх /Тл/Е/Ор</td></tr>
<tr><td>1. Висока швидкість
2.Точність
3. Сумісний з більшістю ПК</td><td></td><td></td></tr>
<tr><td colspan="3">Якість: стандарти, нормативи, параметри тестування тощо</td></tr>
</table>

	Стандартизація відповідно до ДСТУ,ISO. Регламентується НД, СРМ.
	Пакування відсутнє.
	Марка: назва організації-розробника + назва товару
Потенційний товар буде захищено від копіювання: патентування, сертифікати відповідності.	

Таблиця 5.20 — Визначення меж встановлення ціни (із розрахунку на 1 рік користування)

Рівень цін на товари-замінники	Рівень цін на товари-аналоги	Рівень доходів цільової групи споживачів	Верхня та нижня межі встановлення ціни на товар/послугу
>5000 грн.	10000-20000грн.	15-25 тис. грн	4500-7500грн

Таблиця 5.21 — Формування системи збуту

№ п/п	Специфіка закупівельної поведінки цільових клієнтів	Функції збуту, які має виконувати постачальник товару	Глибина каналу збуту	Оптимальна система збуту
	Закупівля можлива після інструктажу та презентації ПЗ, з детальним роз’ясненням можливостей та обмежень програми.	Маркетингові дослідження, обслуговування проданих товарів, Прийняття на себе ризику торгових угод	Канал нульового рівня (прямий маркетинг)	Торгівля через веб-сайт, що належить виробнику.

Таблиця 5.22 — Концепція маркетингових комунікацій

№ п/п	Специфіка поведінки цільових клієнтів	Канали комунікацій, якими користуються цільові клієнти	Ключові позиції, обрані для позиціонування	Завдання рекламного повідомлення	Концепція рекламного звернення
1.	Довіра до технологічних інженерів, що рекомендують товар	Державні установи	Налагодження контактів з будь-якими установами, які мають потребу ідентифікації користувачів.	Донести основну ідею, цінову політику, якість	Продемонструвати переваги перед існуючим товарами

Безперечно є можливість отримати прибуток з даного продукту. Убезпечення важливої інформації завжди було затребуване в різних сферах діяльності, тому рентабельність на ринку дуже висока.

Існують перспективи на впровадження з огляду на потенційні групи клієнтів як в нашій країні, так і за кордоном. Бар’єри входження на ринок відносно високі, через високу конкуренцію, але через низьку вартість програмного забезпечення та його встановлення, дозволить залучити велику кількість користувачів. Тому, конкурентоспроможність проекту достатньо висока.

ВИСНОВКИ

У даній дисертаційній роботі проведено аналіз характеристик клавіатурного почерку, існуючих методів, моделей та засобів отримання КП користувачів. Запропоновано визначення отримання клавіатурного почерку користувача, що відрізняється від існуючих тим, що розпізнавання КП відбувається не тільки за парольним методом, а також за вільним, згенерованим текстом і отриманий шаблон не залежить від тексту і порядку символів набраним користувачем, що надає змогу використовувати даний метод в постійному прихованому моніторингу з метою виявлення підміни законного користувача та визначення відхилення користувача від нормального психофізичного стану.

Розглянуто аналітичну модель, що дозволяє порівнювати отриманий шаблон при реєстрації та поточний шаблон користувача. Використано алгоритм отримання шаблону клавіатурного почерку користувача за часом утримання конкретної клавіші, часом між натисненням клавіш, часом утримання клавіш та часом між натисненням клавіш. Також побудовано алгоритм отримання шаблону, авторизації, аутентифікації та постійного прихованого моніторингу по клавіатурному почерку.

Реалізовано програму, шо використовує алгоритми і методи розпізнавання клавіатурного почерку, які можуть використовуватися в інформаційних системах.

СПИСОК ЛІТЕРАТУРИ

1. Guven, A. Understanding users' keystroke patterns for computer access security. / A. Guven, I. Sogukpinar. // Computers & Security, 2003. – 695-706 с.
2. Butsch, R. Eye movements and the eye-hand span in typewriting. / R. Butsch // Journal of Educational Psychology, 1932. – 104-121 с.
3. User authentication system by using keystroke dynamics/ Dong – Her Shih, Tse – Chun Lin// Department of Information Management, National Yunlin University of Science and Technology, 2008.
4. Electronic capture and analysis of fraudulent behavioral patterns: an application to identity fraud/ Benjamin Ngugi// A Dissertation Submitted to the Faculty of New Jersey Institute of Technology In Partial Fulfillment of the Requirements for the Degree of Doctor of Philosophy in Information Systems, Department of Information Systems, 2005.
5. Umphress, D. Identity Verification through Keyboard CharacteristicsT/ G. Williams// International Journal of Man - Machine Studies, 2005. – 263-273 с.
6. Bleha, S. Computer – access security system using keystroke dynamics/C. Slivinsky, B. Hussein// IEEE Transactions on Pattern Analysis and Machine Intelligence, 1990. – 1217-1222 с.
7. Brown, M. User identification via keystroke characteristics of typed names using neural networks// S. J. Rogers// International Journal of Man – Machine Studies, 1998. – 999-1014 с.
8. Прохоров А. Пропуск, который всегда с собой: Снова о характеристиках биометрических устройствах контроля доступа/ КомпютерПресс - №3.-2001. [Електронний ресурс]. – Режим доступу до ресурсу: http://www.compress.ru/Archive/CP/2001/3/5/10.html.
9. DataLossDB Open security foundation [Електронний ресурс]. – Режим доступу до ресурсу: http://www.datalossdb.org/
10. NtDelayExecution - RealCoding [Електронний ресурс]. – Режим доступу до

ресурсу: http://forums.realcoding.net/lofiversion/index.php/tl6146.html.

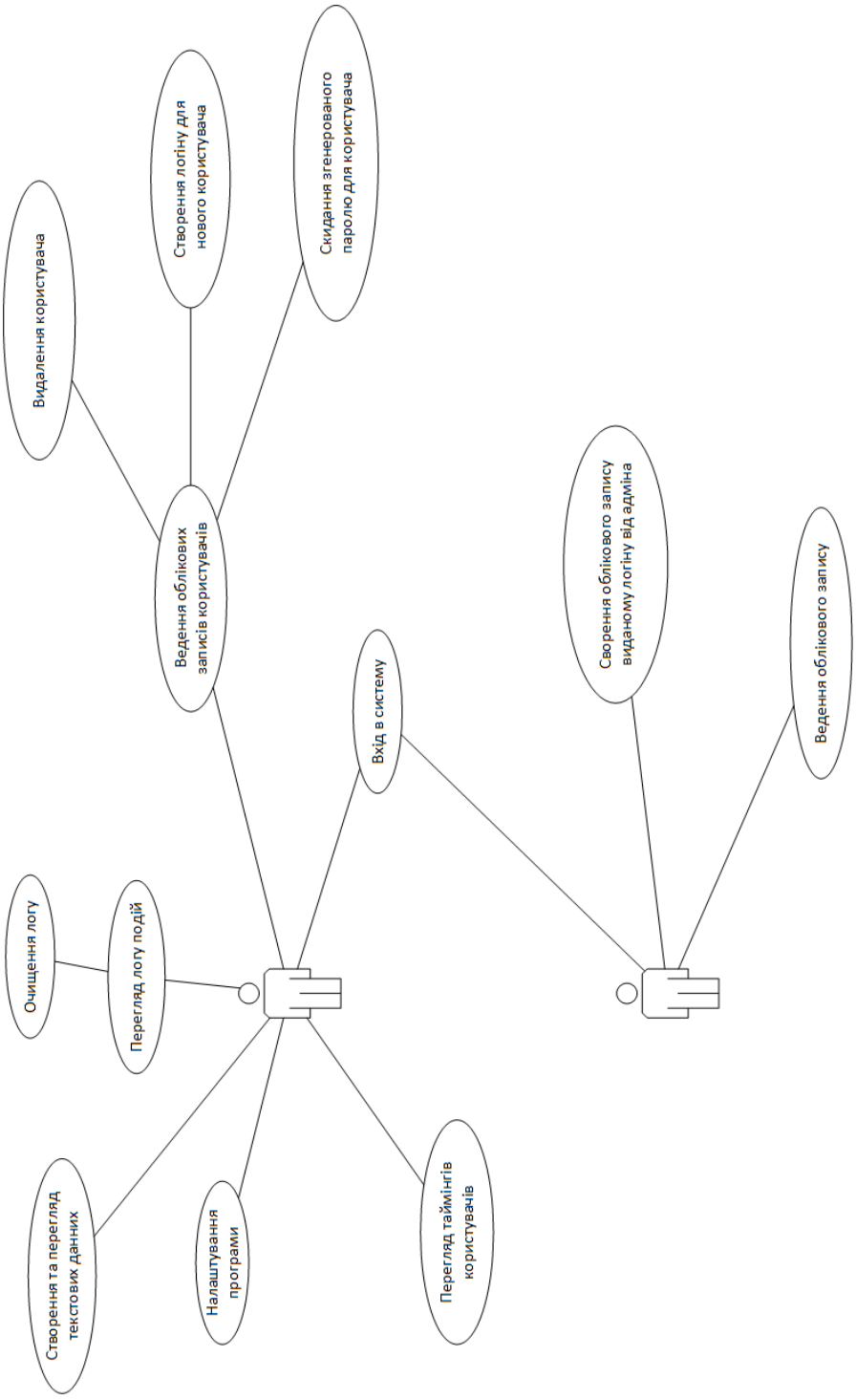
11. Sadoun, B. Verification of Computer Users Using Keystroke Dynamics. / M. S. Obaidat, B. Sadoun /IEEE Transactions on Systems, Man, and Cybernetics - Part B: Cybernetics, 1997. – 261-269 с.
12. Salthouse, T. A. Anticipatory processing in transcription typing. / T. A. Salthouse // J. Appl. Psychol., – 264-271 с.
13. Umphress, D. Identity verification through keyboard characteristics. / D. Umphress, G. Williams // International Journal of Man-Machine Studies, 1985, – 263-273.
14. Yong, S. Weightless Neural Networks for Typing Biometrics Authentication. / S. Yong, W. K. Lai, G. Goghill // In Knowledge-Based Intelligent Information and Engineering Systems, volume 3214 of Lecture Notes in Computer Science, 2004. – 284-293.
15. Сидоркина И.Г., Савинов А.Н. Три алгоритма управления доступом к КСИИ на основе распознавания клавиатурного почерка оператора // Вестник Чувашского университета, 2013. – 293-301 с.
16. Araujo, L., Sucupira, L.H.R., J., Lizarraga, M., Ling, L. & Yabu-Uti, J. User authentication through typing biometrics features, IEEE Transactions on Signal Processing 53, 2005. – 851-855 с.
17. Bergadano, F., Gunetti, D. & Picardi, C. User authentication through keystroke dynamics, ACM Transactions on Information and System Security (TISSEC), 2002. – 367-397.
18. Conklin, A., Dietrich, G. & Walz, D. . Password-based authentication: A system perspective, Proceedings of the 37th Hawaii International Conference on System Sciences, Hawaii, 2004.
19. Crawford, H. (n.d.). Keystroke dynamics: Characteristics and opportunities, Privacy Security and Trust (PST), Eighth Annual International Conference on, IEEE, 2010. – 205-212 с.
20. Широчин В.П. Динамічна характеристика аутентифікація на основі аналізу клавіатурного почерку / Кулик А.В., Марченко В.В .// Вісник національного

технічного університету України "Інформатика, управління та обчислювальна техніка" №32, Київ, 2003.

21. Абашин, В.Г. Автоматизація процесу визначення психофізіологічного стану оператора автоматизованого робочого місця в АСУТП / Орел, - ОрёлГТУ, 2007.
22. Brown, M. User identification via keystroke characteristics of typed names using neural networks // S. J. Rogers // International Journal of Man - Machine Studies, 1998. – 999-1014 с.
23. Эйкхофф, П. Основы идентификации систем управления/ М.: Мир, 2010. – 243 с.

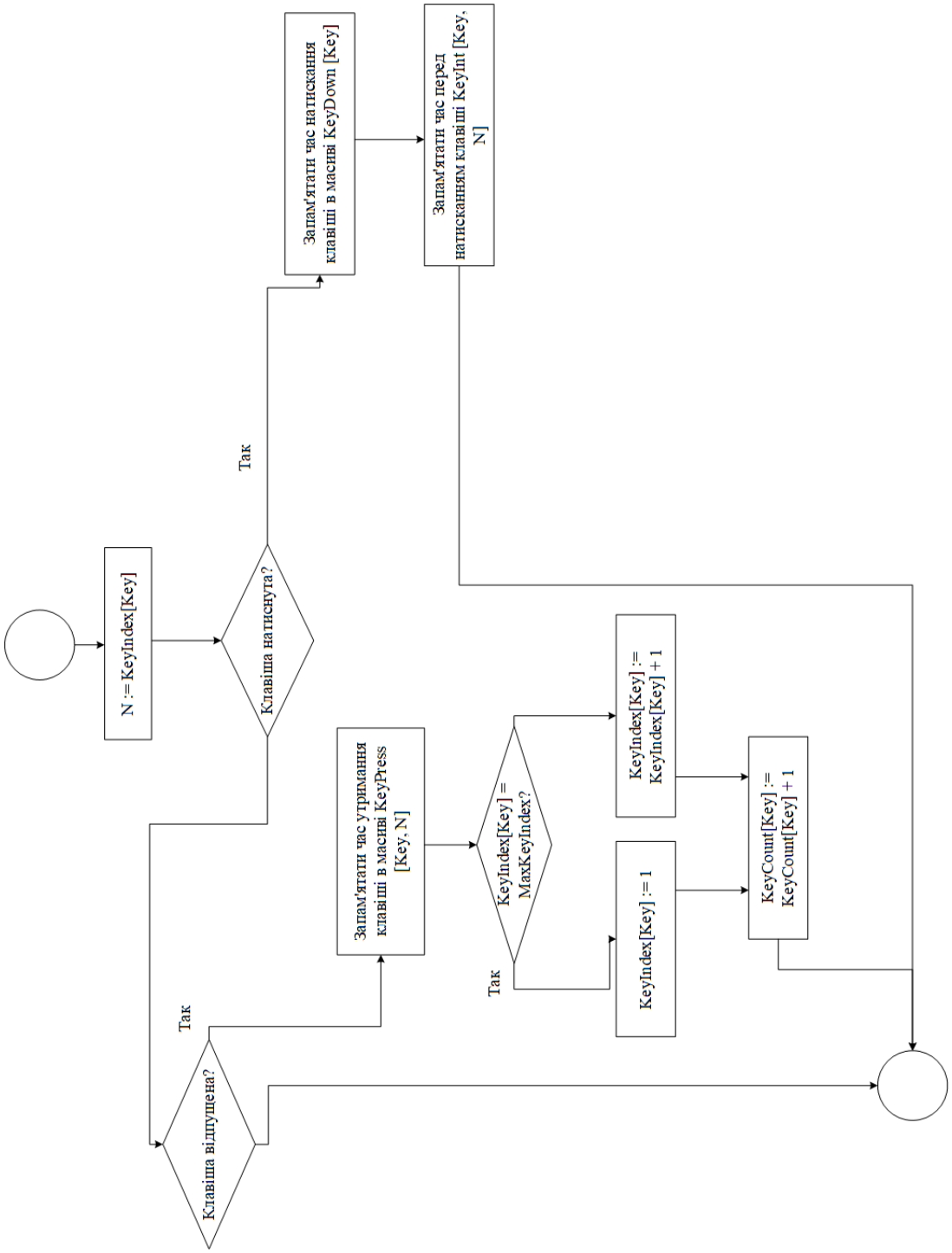
ДОДАТОК А

Ідентифікація користувачів на основі клавіатурного почерку. Сценарій використання системи.



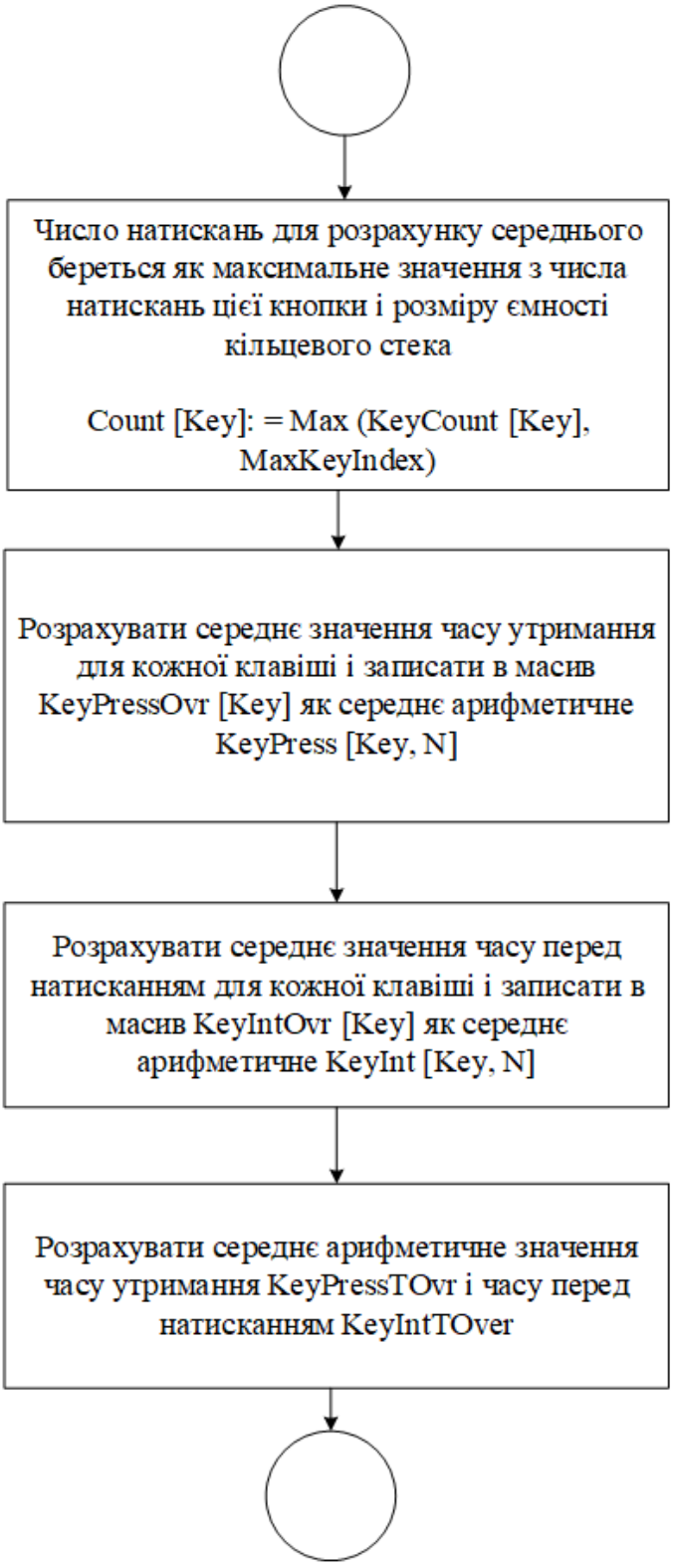
ДОДАТОК Б

Ідентифікація користувачів на основі клавіатурного почерку. Запис даних в кільцевий стек.



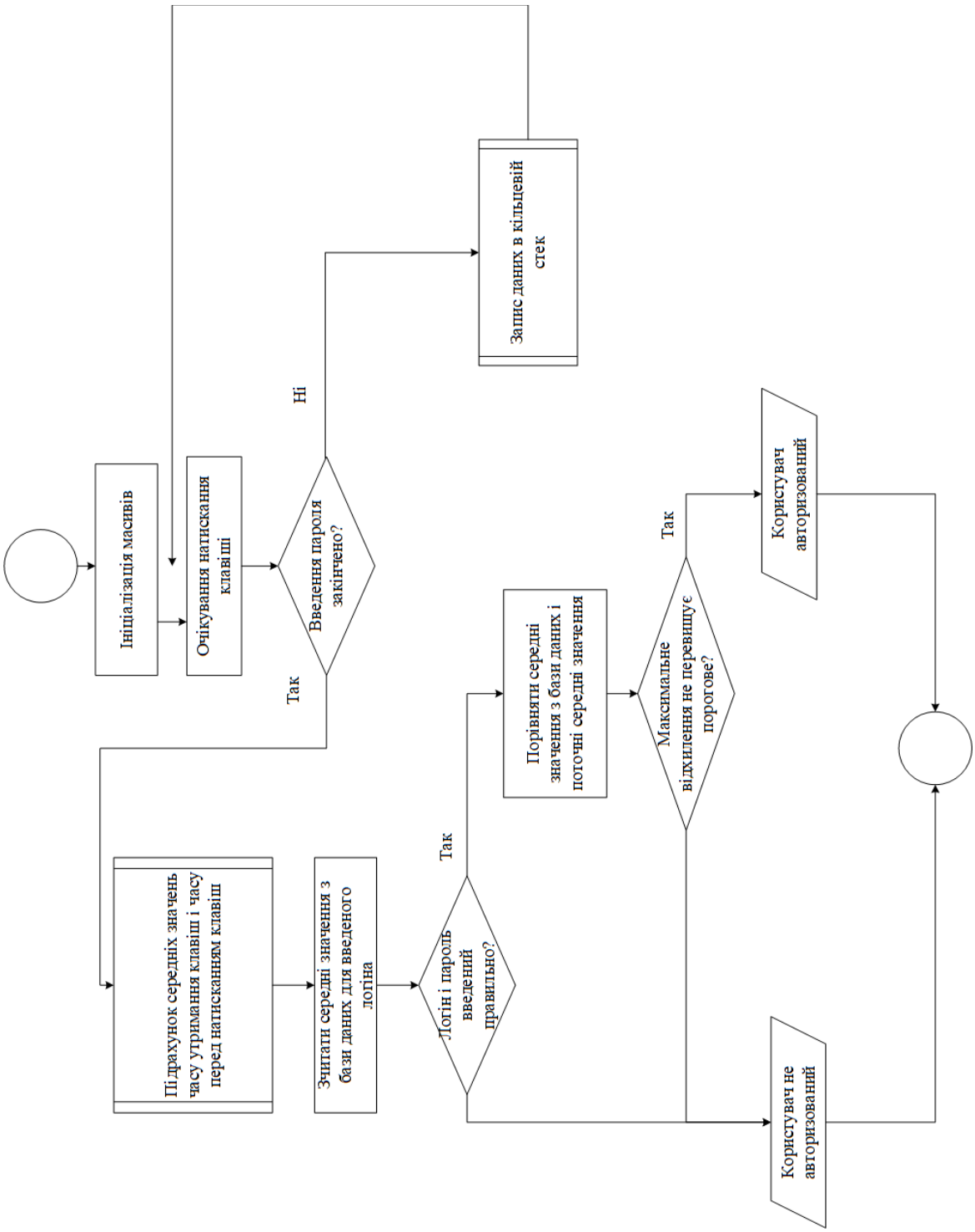
ДОДАТОК В

Ідентифікація користувачів на основі клавіатурного почерку. Отримання шаблону.



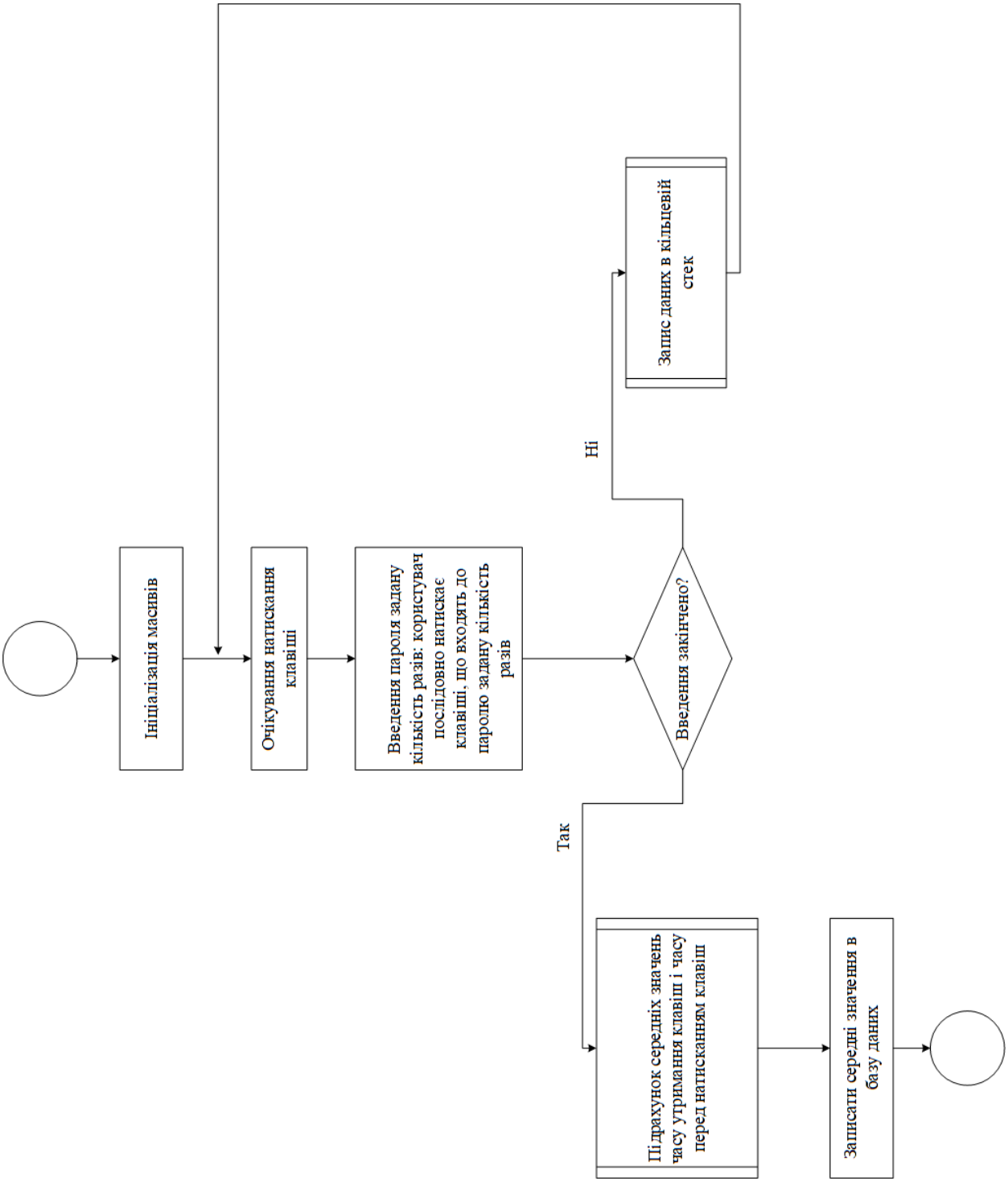
ДОДАТОК Г

Ідентифікація користувачів на основі клавіатурного почерку. Алгоритм авторизації.



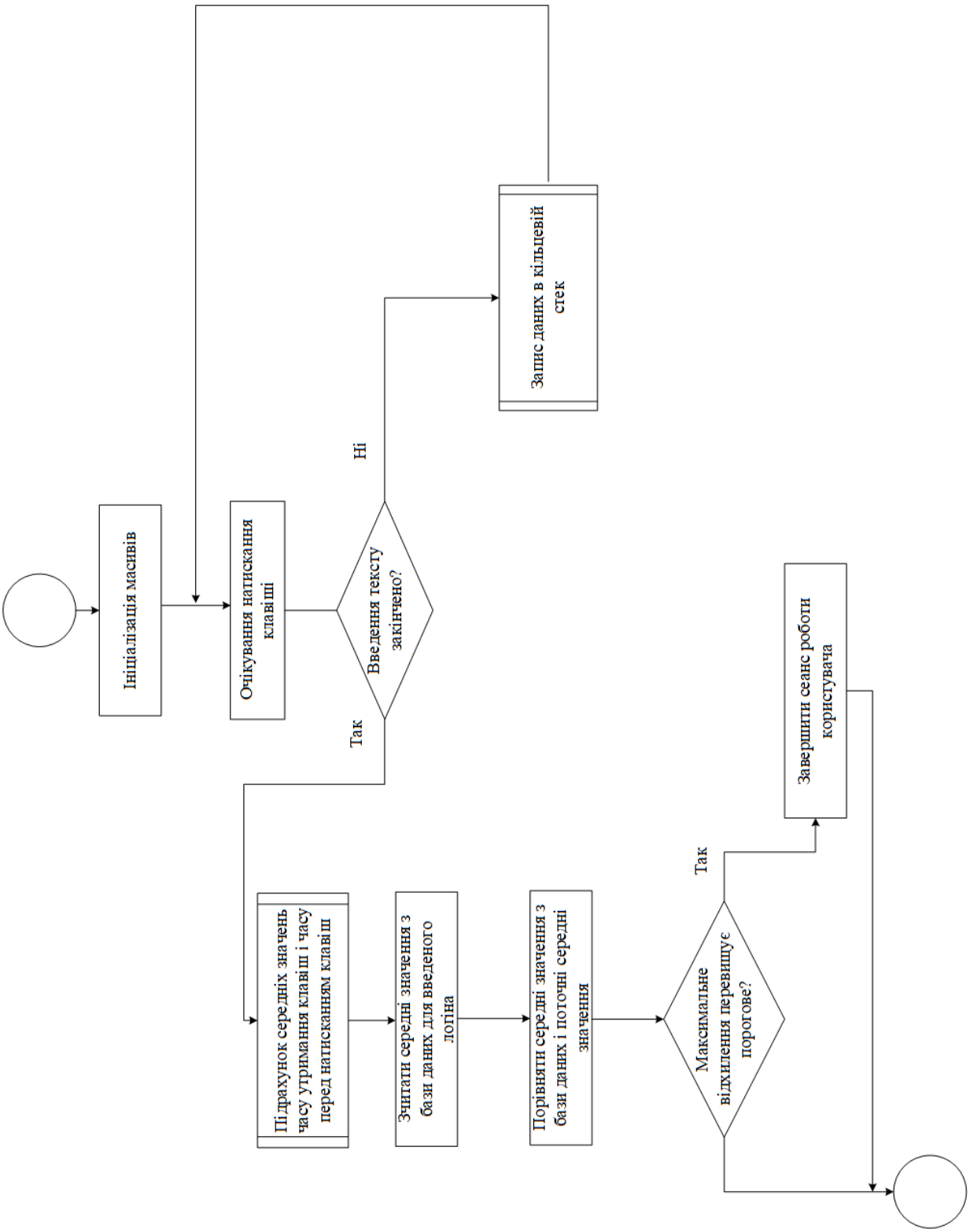
ДОДАТОК Д

Ідентифікація користувачів на основі клавіатурного почерку. Алгоритм аутентифікації.



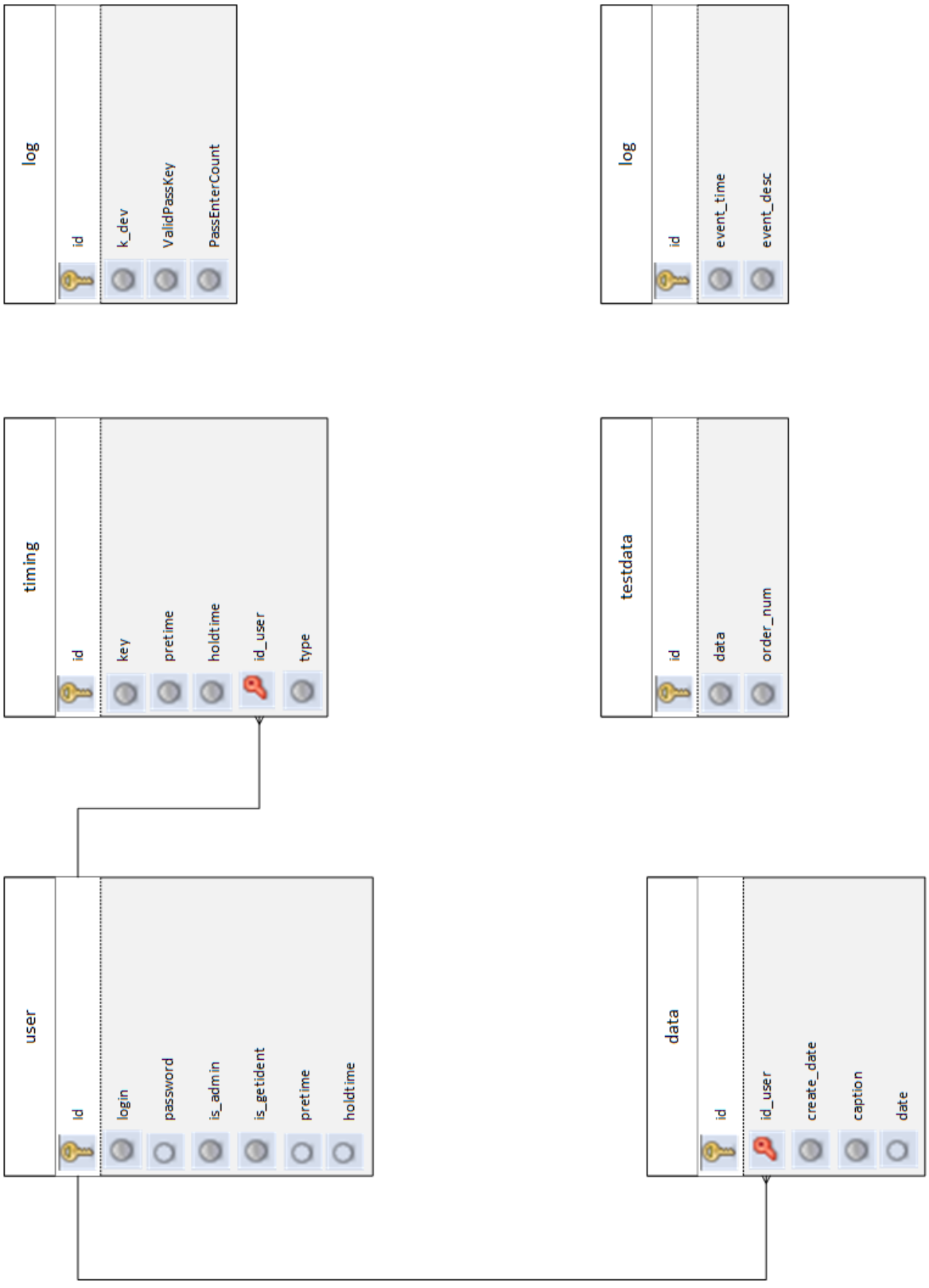
ДОДАТОК Е

Ідентифікація користувачів на основі клавіатурного почерку. Алгоритм моніторингу.



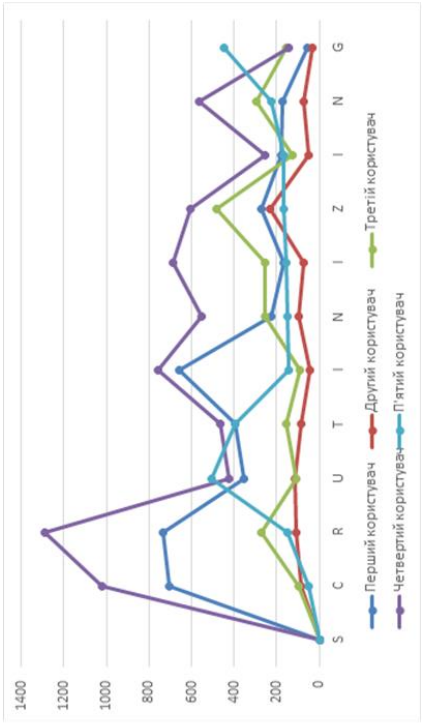
ДОДАТОК Ж

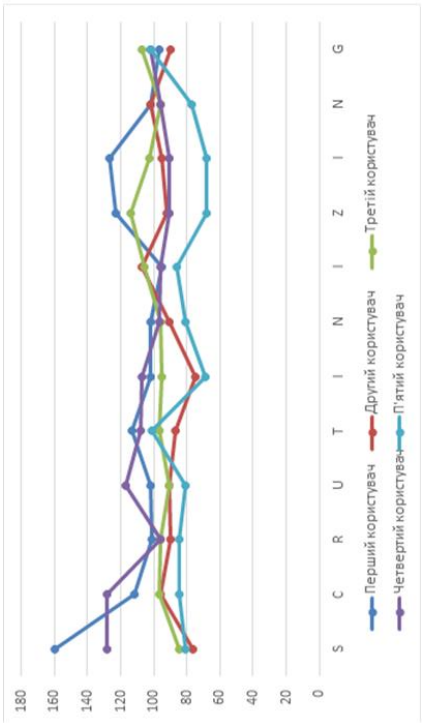
Ідентифікація користувачів на основі клавіатурного почерку. Схема бази даних.



ДОДАТОК К

Ідентифікація користувачів на основі клавіатурного почерку. Порівняння часу утримання клавіші та часу між натисканням клавіш у п’яти користувачів.





ДОДАТОК Л

Основна частина коду програми

```
const
  MAX_SER_LEN = 20; // обсяг статистики натиснень кожної клавіші, що зберігається
  MAX_KEY_CODE = 230; // максимальне значення коду
  // типи таймінгів
  TT_LOGIN = 0; // введення логіну
  TT_TEXT = 1;  // введення тексту
  MIN_PASS_LENGTH = 5; // мінімальна довжина паролю
  VAR_PASS_LENGTH = 3; // можливі коливання довжини пароля
var
  // допустимі символи для пароля
  ValidPassKey: string;
  EnterPassCount: integer;
  // ідентифікатор поточного користувача
  UserID: integer;
  // час попереднього натискання
  PredPress: TTime;
  // масиви для вимірювань;
  KeyDown : array[1..MAX_KEY_CODE] of TTime; // момент натискання клавіші
  KeyIndex: array[1..MAX_KEY_CODE] of integer; // індекс для зберігання серії даних і організації кільцевого стека зберігання
  KeyCount: array[1..MAX_KEY_CODE] of integer; // лічильник натискань.
  KeyPress: array[1..MAX_KEY_CODE,1..MAX_SER_LEN] of integer; // час утримання клавіші
  KeyInt: array[1..MAX_KEY_CODE,1..MAX_SER_LEN] of integer; // час перед натисканням клавіші (між натисканнями)
// поточна статистика
  KeyPressOvr: array[1..MAX_KEY_CODE] of integer; // середній час утримання клавіші
  KeyIntOvr: array[1..MAX_KEY_CODE] of integer; // середній час перед натисканням клавіші (між натисканнями)
  KeyPressTOvr: integer; // середній час утримання для всіх клавіші
  KeyIntTOvr: integer; // середній час перед натисканням клавіші (між натисканнями) для всіх клавіш
  // статистика авторизованого користувача
```

```
  UserKeyPressOvr: array[1..MAX_KEY_CODE] of integer; // середній час утримання клавіші
  UserKeyIntOvr: array[1..MAX_KEY_CODE] of integer; // середній час перед натисканням клавіші (між натисканнями)
  UserKeyPressTOvr: integer; // середній час утримання для всіх клавіші
  UserKeyIntTOvr: integer; // середній час перед натисканням клавіші (між натисканнями) для всіх клавіш

procedure frmTiming_OnShow (Sender: TObject; Action: string);
begin
  // оновити відображення таблиці таймінгу
  frmTiming.btnTiming.Click;
end;

// підключити для нижченаведених контролів облік таймінгу набору тексту
procedure frmDataEdit_Memo1_OnKeyUp (Sender: TObject; var Key: Word; Shift, Alt, Ctrl: boolean);
begin
  Sign_OnKeyUp (Key);
end;

procedure frmDataEdit_Memo1_OnKeyDown (Sender: TObject; var Key: Word; Shift, Alt, Ctrl: boolean);
begin
  Sign_OnKeyDown (Key);
end;

procedure frmGetParam_memEnterData_OnKeyUp (Sender: TObject; var Key: Word; Shift, Alt, Ctrl: boolean);
begin
  Sign_OnKeyUp (Key);
end;

procedure frmGetParam_memEnterData_OnKeyDown (Sender: TObject; var Key: Word; Shift, Alt, Ctrl: boolean);
begin
  Sign_OnKeyDown (Key);
end;
```

```
procedure frmChangePass_edtEnterPass_OnKeyUp (Sender: TObject; var Key: Word;
Shift, Alt, Ctrl: boolean);
begin
  Sign_OnKeyUp (Key);
end;

procedure frmChangePass_edtEnterPass_OnKeyDown (Sender: TObject; var Key: Word;
Shift, Alt, Ctrl: boolean);
begin
  Sign_OnKeyDown (Key);
end;

procedure frmLogin_edtPassword_OnKeyUp (Sender: TObject; var Key: Word; Shift, Alt,
Ctrl: boolean);
begin
  Sign_OnKeyUp (Key);
end;

procedure frmLogin_edtPassword_OnKeyDown (Sender: TObject; var Key: Word; Shift,
Alt, Ctrl: boolean);
begin
  Sign_OnKeyDown (Key);
end;

// для зняття сигнатур
// натискання клавіші
procedure Sign_OnKeyDown (Key: Word);
var
  tmpPressTime:TTime;
  tmpInterval: integer;
begin
  if (Key >= 1) and (Key <= MAX_KEY_CODE) then
  begin
    tmpPressTime := Time();
    KeyDown[Key] := tmpPressTime;
    inc( KeyCount[Key] );
    try
```

```
      // іноді інтервал може стати занадто великим ...
      tmpInterval := MilliSecondsBetween(tmpPressTime,PredPress); // інтервал між
натисканням, мілісекунд
    except
      tmpInterval := 50;
    end;
    KeyInt[Key,KeyIndex[Key]] := tmpInterval;
    PredPress := tmpPressTime;
  end;
end;

// для зняття сигнатур
// відпускання клавіші
procedure Sign_OnKeyUp (Key: Word);
var
  tmpInterval: integer;
begin
  if (Key >= 1) and (Key <= MAX_KEY_CODE) then
  begin
    tmpInterval := MilliSecondsBetween(Time(),KeyDown[Key]); // час натискання
(утримання) клавіші
    KeyPress[Key,KeyIndex[Key]] := tmpInterval;
    inc(KeyIndex[Key]);
    if KeyIndex[Key] > MAX_SER_LEN then
      KeyIndex[Key] := 1;
  end;
end;

// обчислення статистики
// розраховується середньоарифметичне:
// 1. час до одним натисненням
// 2. час утримання певної клавіші
// 3. час до натискання клавіші
// 4. час утримання клавіші
procedure CalcStatistic;
var
  tmpKey:integer;
  i: integer;
```

```
  tmpKeyCount: integer;
begin
  tmpKeyCount := 0;
  KeyPressTOvr := 0;
  // час утримання клавіші
  for tmpKey := 1 to MAX_KEY_CODE do
  begin
    KeyPressOvr[tmpKey] := 0;
    KeyIntOvr[tmpKey] := 0;
    if KeyCount[tmpKey] > 0 then
    begin
      i := 1;
      while (i <= KeyCount[tmpKey]) and (i<=MAX_SER_LEN) do
      begin
        KeyPressOvr[tmpKey] := KeyPressOvr[tmpKey] + KeyPress[tmpKey,i];
        KeyIntOvr[tmpKey] := KeyIntOvr[tmpKey] + KeyInt[tmpKey,i];
        inc(i);
      end;
      dec(i); // скільки серій підраховано
      KeyPressOvr[tmpKey] := KeyPressOvr[tmpKey] div i; //середнє арифметичне для
даної клавіші
      KeyIntOvr[tmpKey] := KeyIntOvr[tmpKey] div i;
      //
      KeyPressTOvr := KeyPressTOvr + KeyPressOvr[tmpKey];
      KeyIntTOvr := KeyIntTOvr + KeyIntOvr[tmpKey];
      inc( tmpKeyCount );
    end;
  end;
  if tmpKeyCount<>0 then
  begin
    KeyPressTOvr := KeyPressTOvr div tmpKeyCount; // середній час по всіх клавішах
    KeyIntTOvr := KeyIntTOvr div tmpKeyCount; // середній час по всіх клавішах
  end;
end;

// збереження статистики в базу
// дані зберігаються в таблиці "timing" і "user"
procedure SaveTiming(AType:integer);
```

```
var
  tmpKey:integer;
begin
  CalcStatistic;
  // записуємо базову статистику
  SQLExecute('update user set pretime = '+inttostr(KeyIntTOvr)+', holdtime =
'+inttostr(KeyPressTOvr)+' where id = '+inttostr(UserID));
  // стираємо розширену статистику
  SQLExecute('delete from timing where id_user = '+inttostr(UserID)+' and type =
'+inttostr(AType));
  // додаємо розширену статистику
  for tmpKey := 1 to MAX_KEY_CODE do
  begin
    if (KeyPressOvr[tmpKey]<>0) and (KeyIntOvr[tmpKey]<>0) then
    begin
      SQLExecute('insert into timing (key,pretime,holdtime,id_user,type) '+
      'values
('+inttostr(tmpKey)+','+inttostr(KeyIntOvr[tmpKey])+','+inttostr(KeyPressOvr[tmpKey])+
','+inttostr(UserID)+','+inttostr(AType)+')');
    end
  end;
end;

// завантаження таймінгу для поточного користувача
// дані завантажуються з таблиць "timing" і "user"
procedure LoadTiming(AType:integer);
var
  tmpKey:integer;
  tmpDataSet: TDataSet;
begin
  for tmpKey := 1 to MAX_KEY_CODE do
  begin
    UserKeyPressOvr[ tmpKey ] := 0;
    UserKeyIntOvr[ tmpKey ] := 0;
  end;
  // перевіряємо загальні показники і розширені показники по окремих клавішах
  UserKeyPressTOvr := SQLExecute('select coalesce(holdtime,0) from user where id =
'+inttostr(UserID));
```

```
  UserKeyIntTOvr := SQLExecute('select coalesce(pretime,0) from user where id =
'+inttostr(UserID));
  //
  SQLQuery('select * from timing where id_user = '+inttostr(UserID)+' and type =
'+inttostr(AType),tmpDataSet);
  while not tmpDataSet.EOF do
  begin
    UserKeyPressOvr[ tmpDataSet.FieldByName('key').asInteger ] :=
tmpDataSet.FieldByName('holdtime').asInteger;
    UserKeyIntOvr[ tmpDataSet.FieldByName('key').asInteger ] :=
tmpDataSet.FieldByName('pretime').asInteger;
    tmpDataSet.Next;
  end;
  tmpDataSet.Free;
end;

// ключова функція програми: порівняння сигнатур
// окремо порівнюються сигнатури для пароля і сигнатури при наборі тексту, так як
склад символів в цих випадках різна,
// що може відбиватися на динаміці набору тексту.
function CheckSign(AType:integer):boolean;
var
  tmpDiv: double;
  tmpDiv2: double;
  tmpDivMax: double; //максимальне відхилення
  tmpCount: integer;
  tmpKey:integer;
begin
 // розрахувати середні значення по поточним відомостями про дії користувача
  CalcStatistic;
 // в процесі порівняння знаходимо максимальне відхилення характеристик
  tmpDivMax := 0;
  AddToLog('Перевірка сигнатур. Тип перевірки: '+inttostr(AType));
  tmpDiv := abs(UserKeyPressTOvr - KeyPressTOvr) / UserKeyPressTOvr;
  if tmpDiv > tmpDivMax then
    tmpDivMax := tmpDiv;
  AddToLog('Відхилення по утриманню: '+FloatToStr(tmpDiv));
  tmpDiv := abs(UserKeyIntTOvr - KeyIntTOvr) / UserKeyIntTOvr;
```

```
  if tmpDiv > tmpDivMax then
    tmpDivMax := tmpDiv;
  AddToLog('Відхилення по темпу: '+FloatToStr(tmpDiv));
  //
  tmpDiv := 0;
  tmpDiv2 := 0;
  tmpCount := 0;
  for tmpKey := 1 to MAX_KEY_CODE do
  begin
    // пропускаємо порожні рядки в таблиці статистики
    if (KeyPressOvr[tmpKey]<>0) and (KeyIntOvr[tmpKey]<>0) and
(UserKeyPressOvr[tmpKey]<>0) and (UserKeyIntOvr[tmpKey]<>0) then
    begin
      tmpDiv := tmpDiv + abs(UserKeyIntOvr[tmpKey] - KeyIntOvr[tmpKey]) /
UserKeyIntOvr[tmpKey];
      tmpDiv2 := tmpDiv2 + abs(UserKeyPressOvr[tmpKey] - KeyPressOvr[tmpKey]) /
UserKeyPressOvr[tmpKey];
      inc(tmpCount);
    end
  end;
  if tmpCount>0 then
  begin
    tmpDiv := tmpDiv / tmpCount;
    tmpDiv2 := tmpDiv2 / tmpCount;
  end;
  AddToLog('Середнє відхилення по утриманню по клавішах: '+FloatToStr(tmpDiv));
  AddToLog('Середнє відхилення по темпу по клавішах: '+FloatToStr(tmpDiv2));
  if tmpDiv > tmpDivMax then
    tmpDivMax := tmpDiv;
  if tmpDiv2 > tmpDivMax then
    tmpDivMax := tmpDiv2;
  // порівнюємо максимальне відхилення з максимально допустимим відхиленням,
яке зберігається в настройках
  result := tmpDivMax < SQLExecute('select coalesce(k_dev,0) from param');
  if not result then
    AddToLog('Виявлено невідповідність психомоторних характеристик поточного
користувача і реєстраційних даних');
end;
```

```
// ініціалізація системи аналізу таймингов
procedure InitTimings;
var
  i:integer;
  j:integer;
begin
  for i:=1 to MAX_KEY_CODE do
  begin
    KeyDown[i] := 0;
    KeyIndex[i] := 1;
    KeyCount[i] := 0;
    for j:=1 to MAX_SER_LEN do
    begin
      KeyPress[i,j] := 0;
      KeyInt[i,j] := 0;
    end;
  end;
  PredPress:=Now();
end;

// генерація випадкового пароля
procedure frmChangePass_brnCreatePass_OnClick (Sender: TObject; var Cancel:
boolean);
var
  tmpPassLength: integer;
  tmpPass: string;
  tmpValidKeys: string;
  i: integer;
  j: integer;
begin
  // довжина пароля буде не менше мінімальної, але відрізнятися на випадкову
величину від 0 до VAR_PASS_LENGTH
  tmpPassLength := MIN_PASS_LENGTH +
trunc(Random()*(VAR_PASS_LENGTH+1));
  // для побудови пароля береться набір допустимих символів, який зберігається в
настройках, в таблиці "param"
  tmpValidKeys := SQLExecute('select coalesce(ValidPasskey,"!") from param');
```

```
  tmpPass := '';
  for i:=1 to tmpPassLength do
  begin
    tmpPass := tmpPass + tmpValidKeys[ trunc(Random()*length(tmpValidKeys))+1 ];
  end;
  frmChangePass.edtPassword.Text := tmpPass;
  frmChangePass.edtEnterPass.Text := '';
  frmChangePass.btnConfirm.Enabled := true;
  // підготувати запис таймінгу
  InitTimings;
end;

// контроль при збереження даних
procedure frmDataEdit_Button1_OnAfterClick (Sender: TObject);
begin
  if not CheckSign(TT_TEXT) then
  begin
    ShowMessage('Виявлено розбіжність психомоторних характеристик поточного користувача і реєстраційних даних. Доступ до програми заблокований. Будь ласка, пройдіть повторну авторизацію!');
    frmMain.Close;
  end;
end;

// закриття основної форми
procedure frmMain_OnClose (Sender: TObject; Action: string);
begin
  // закрити програму
  frmLogin.close;
end;

// для нового запису підставляється ID поточного користувача
procedure frmDataEdit_OnShow (Sender: TObject; Action: string);
begin
  if Action = 'NewRecord' then
  begin
    frmDataEdit.cmbUser.dbItemID := UserID;
  end;
```

```
end;

// відображення головної форми
procedure frmMain_OnShow (Sender: TObject; Action: string);
begin
  frmMain.tgrData.dbUpdate;
  // завантажити сигнатури користувача
  LoadTiming(TT_TEXT);
end;

// перевірка написаного тексту і збереження сигнатури набору тексту
procedure frmGetParam_btnOK_OnClick (Sender: TObject; var Cancel: boolean);
begin
  if frmGetParam.memEnterData.Text = frmGetParam.memTestData.Text then
  begin
    SaveTiming(TT_TEXT);
    frmGetParam.ModalResult := mrOK
  end
  else
  begin
    ShowMessage('В процесі набору були допущені помилки. Повторіть набір');
    frmGetParam.memEnterData.Text := '';
    PredPress:=Now();
  end;
end;

// завантаження випадкового тексту для зняття сигнатури
// дані завантажуються з таблиці "testdata"
procedure frmGetParam_OnShow (Sender: TObject; Action: string);
var
  tmpMaxCount: integer;
begin
  frmGetParam.memEnterData.Text := '';
  // загрузить текст
  tmpMaxCount := SQLExecute('select count(*) from testdata');
  frmGetParam.memTestData.Text := SQLExecute('select data from testdata where
order_num = '+inttostr(trunc( Random()*tmpMaxCount + 1 )));
  PredPress:=Now();
```

```
end;

// після кожного набору нового пароля підраховується кількість повторів введення
// потім, при необхідності, викликається вікно зняття сигнатури для набору тексту
procedure frmChangePass_btnConfirm_OnClick (Sender: TObject; var Cancel: boolean);
var
  tmpEnterPassLeft:integer;
begin
  if frmChangePass.edtPassword.Text = frmChangePass.edtEnterPass.Text then
  begin
    frmChangePass.edtEnterPass.Text := '';
    inc(EnterPassCount);
    tmpEnterPassLeft := SQLExecute('select coalesce(PassEnterCount,3) from param') -
EnterPassCount;
    frmChangePass.labEnterPassLeft.Caption := 'Залишилося повторів:
'+inttostr(tmpEnterPassLeft);
    PredPress:=Now();
    if tmpEnterPassLeft = 0 then
    begin
      // збереження сигнатури
      SaveTiming(TT_LOGIN);
      if not frmChangePass.chbGetIdent.Checked then
      begin
        ShowMessage('Вам необхідно пройти додаткову ідентифікацію.');
        if frmGetParam.ShowModal = mrOK then
        begin
          frmChangePass.chbGetIdent.Checked := true;
          frmChangePass.btnConfirm.Enabled := false;
          frmChangePass.btnSave.Enabled := true;
        end
        else
        begin
          frmChangePass.btnClose.Click;
        end;
      end
      else
      begin
        frmChangePass.btnConfirm.Enabled := false;
```

```
        frmChangePass.btnSave.Enabled := true;
      end;
    end;
  end
  else
  begin
    ShowMessage('При наборі пароля допущена помилка!');
  end;
end;

//відкриття форми для редагування параметрів
procedure frmLog_btnParamEdit_OnClick (Sender: TObject; var Cancel: boolean);
var
  tmpID:integer;
begin
  // отримання ідентифікатора запису з настройками, якщо налаштувань ще немає, то
ID = -1
  tmpID := SQLExecute('select coalesce(id,-1) from param');
  // відобразити форму редагування налаштувань
  frmParamEdit.ShowRecord('param',tmpID);
end;

//ініціалізація форми створення нового пароля
procedure frmChangePass_OnShow (Sender: TObject; Action: string);
begin
  frmChangePass.edtEnterPass.Text := '';
  frmChangePass.btnSave.Enabled := false;
  frmChangePass.btnConfirm.Enabled := false;
  frmChangePass.edtEnterPass.SetFocus;
  EnterPassCount :=0;
  frmChangePass.labEnterPassLeft.Caption := 'Залишилося повторів:
'+inttostr(SQLExecute('select coalesce(PassEnterCount,3) from param') -
EnterPassCount);
end;

// скидання пароля користувача
procedure frmUserEdit_btnClearPassword_OnClick (Sender: TObject; var Cancel:
boolean);
```

```
begin
  frmUserEdit.edtPassword.Clear;
end;

// додавання даних в журнал (таблиця "log")
procedure AddToLog(AText:string);
begin
  SQLExecute('insert into log (event_time,event_desc) values ('"+FormatDateTime('yyyy-
mm-dd hh:mm:ss.nnn',Now())+"',"'+AText+"')');
end;

// очищення лога
procedure frmLog_btnClearLog_OnClick (Sender: TObject; var Cancel: boolean);
begin
  //видаляємо всі дані з логу
  SQLExecute('delete from log');
  // оновлюємо таблицю
  frmLog.tgrLog.dbUpdate;
end;

// оновлення таблиці відображення даних з логу
procedure frmLog_OnShow (Sender: TObject; Action: string);
begin
  frmLog.tgrLog.dbUpdate;
end;

// закриття перегляду логу - основного вікна для адміна
procedure frmLog_OnClose (Sender: TObject; Action: string);
begin
  // завершити роботу програми
  frmLogin.close;
end;

// запуск головної форми додатка - вікна логіна
procedure frmLogin_OnShow (Sender: TObject; Action: string);
begin
  AddToLog('Запуск програми');
  // при першому запуску, коли база порожня, додати адміна
```

```
  if SQLExecute('select count(*) from user where is_admin = 1')=0 then
  begin
    SQLExecute('insert into user (login,password,is_admin) values ("admin","admin",1)');
  end;
  InitTimings;
end;

// авторизація користувача
procedure frmLogin_btnLogin_OnClick (Sender: TObject; var Cancel: boolean);
var
  tmpDataSet:TDataset;
begin
  SQLQuery('select * from user where login = "'+frmLogin.edtLogin.Text+'" ',tmpDataset);
  if tmpDataSet.eof then
  begin
    AddToLog('Невдала спроба авторизації. Невірний логін');
    ShowMessage('Невірний логін')
  end
  else
  begin
    // запам'ятовуємо ідентифікатор поточного користувача, стане в нагоді при
додаванні записів в щоденник
    UserID := tmpDataset.FieldByName('id').asInteger;
    if tmpDataset.FieldByName('password').isNull then
    begin // новий користувач, доданий адміном. Він повинен придумати собі пароль
      ShowMessage('Ваш обліковий запис вимагає установки пароля ');
      frmChangePass.ShowRecord('user',tmpDataset.FieldByName('id').asInteger);
    end
    else
    begin // у користувача є пароль
      if tmpDataset.FieldByName('password').asString <> frmLogin.edtPassword.Text then
      begin
        AddToLog('Невдала спроба авторизації. Невірний пароль');
        ShowMessage('Невірний пароль')
      end
      else
      begin
        if tmpDataset.FieldByName('is_admin').asInteger = 1 then
```

```
      begin
        AddToLog('У програму увійшов користувач з правами адміністратора під
логіном "'+frmLogin.edtLogin.Text+'"');
        frmLogin.Hide;
        frmLog.Show;
      end
      else
      begin
        // завантажити сигнатури користувача
        LoadTiming(TT_LOGIN);
        // звичайний користувач - потрібна додаткова перевірка сигнатур
        if CheckSign(TT_LOGIN) then
        begin
          AddToLog('У програму увійшов користувач під логіном
"'+frmLogin.edtLogin.Text+'"');
          frmLogin.Hide;
          frmMain.Show;
        end
        else
        begin
          ShowMessage('Ви не пройшли психомоторну перевірку. В доступі
відмовлено.');
          AddToLog('Користувач "'+frmLogin.edtLogin.Text+'" не пройшов
психомоторну перевірку. В доступі відмовлено');
        end;
      end
    end;
  end;
end;

// зниження похибки вимірювання
// звіт для першого натискання йде від отримання контролом фокуса
procedure frmLogin_edtPassword_OnEnter (Sender: TObject);
begin
  PredPress := now();
end;
```

```
procedure frmDataEdit_Memo1_OnEnter (Sender: TObject);
begin
  PredPress := Now();
end;

begin
// ініціалізація генератора псевдовипадкових чисел
  Randomize;
end.
```

ДОДАТОК М

Ідентифікація користувачів на основі клавіатурного почерку. Акт впровадження програмного забезпечення.

Затверджую:
Директор ТОВ ПМТЗ "АГРО-ДЕТАЛЬ"
_____ Шевчук Н. Р.
«05» _____ 2018р.

АКТ

Впровадження результатів дипломної роботи

Самойленка Олега Станіславовича

«Ідентифікація користувачів на основі клавіатурного почерку»

Комісія в складі: голова комісії Шевчук Н.Р., члени комісії: Гнатюк І.М., Крамаренко Р.П., склали цей акт про те, що результати дипломної роботи студента Національного технічного університету України «Київського політехнічного інституту імені Ігоря Сікорського», представленої на захист дипломного проекту, використані в ТОВ ПМТЗ "АГРО-ДЕТАЛЬ" як програмного забезпечення ідентифікації користувачів на підставі клавіатурного почерку.

У проекті був використаний повний функціонал програмного забезпечення для ідентифікації користувачів на підставі клавіатурного почерку.

Застосування даних результатів дозволило виключити випадки витоку інформації і контроль над складом та роботою співробітників.

Результати дипломної роботи студента Самойленка О.С. мають практичне значення, є актуальними і потребують подальшого розвитку.

Голова комісії: _____ Шевчук Н. Р.

Члени комісії: _____ Гнатюк І.М.

_____ Крамаренко Р.П.